

**Aplicativo para apoyar el proceso de aprendizaje del lenguaje de señas hacia  
un oyente mediante Microsoft Kinect®**

**Robinson Steven Castro Londoño**

**Universidad Piloto de Colombia  
Facultad de Ingeniería de Sistemas  
Bogotá  
2015**

**Aplicativo para apoyar el proceso de aprendizaje del lenguaje de señas hacia  
un oyente mediante Microsoft Kinect®**

**Robinson Steven Castro Londoño**

**Trabajo de grado**

**Oscar Elías Herrera Bedoya**

**Docente programa ingeniería de sistemas**

**Ignacio Hernández Molina**

**Sociólogo, docente programa ingeniería de sistemas**

**Universidad Piloto de Colombia**

**Facultad de Ingeniería de Sistemas**

**Bogotá**

**2015**

**Nota de aceptación**

---

---

---

---

---

---

**Presidente del Jurado**

---

**Jurado**

---

**Jurado**

**Bogotá 1 de junio de 2015**

## **DEDICATORIA**

Quisiera dedicar este trabajo a mis padres Nancy y Carlos, por su apoyo incondicional, sus consejos, su comprensión, el amor, la ayuda en los momentos difíciles, porque ellos me han forjado como persona, mis valores, mis principios, mi carácter, mi empeño y mi perseverancia para conseguir mis metas.

A mi familia y a mi hermano Felipe que de una u otra manera estuvieron pendientes a lo largo de este proceso, acompañándome día a día a lo largo del camino recorrido, dándome toda la ayuda posible y ofreciendo todo su apoyo, se los agradezco mucho.

A mis compañeras y grandes amigas Ale, Kathy y Jhoa, porque siempre me dieron su apoyo cuando lo necesite. A ustedes que me acompañaron y vivieron conmigo lo largo de este proceso, les agradezco la confianza que depositaron en mí, por cada uno de los momentos vividos juntos, por estar en las buenas y las malas, y haber hecho de mi etapa universitaria un trayecto para recordar.

Con todo mi cariño para todas las personas que me ayudan, ayudaron y acompañan, a ellos que hicieron todo lo posible para que pudiera lograr mis objetivos, dándome motivación y brindarme la mano amiga cuando sentía que el camino se hacía estrecho, a todos ustedes, el corazón no deja que me olvide.

## AGRADECIMIENTOS

Agradezco a Dios quién me guio por el buen camino, dándome fuerzas en los momentos más difíciles, para seguir adelante y no desistir cuando se presentaron las adversidades, enseñándome a enfrentar las circunstancias, sin desfallecer en el intento.

A mis padres por haberme forjado como persona, con principios, disciplina y amor, he alcanzado un nuevo logro gracias a ustedes que me motivaron con constancia para culminar esta etapa que inicio hace 5 años.

A los docentes que me acompañaron durante el arduo camino, brindándome siempre su orientación con profesionalismo ético, siendo comprensibles y estrictos a la vez, gracias por formar y afianzar el perfil no solo profesional sino como persona.

"La lengua de signos está llena de plasticidad y belleza y es capaz de crear la magia de la poesía y de envolver a las personas en un mundo onírico lleno de imágenes fantásticas. Sirve para confesarse, para la filosofía, para discutir o hacer el amor. Está llena de fuerza simbólica... El alma que se escapa por sus dedos es para ellos la vida misma".

Oliver Sacks

## CONTENIDO

	Pág.
1. CAPITULO 1.....	16
1.1 Alcance.....	16
1.2 Limites .....	16
1.3 Objetivos .....	17
1.3.1 General.....	17
1.3.2 Específicos .....	17
2. CAPITULO 2.....	18
2.1 Revisión bibliográfica.....	18
2.2 Marco referencial.....	29
2.3 Sistema de hipótesis .....	32
2.3.1 Hipótesis de trabajo.....	32
2.3.2 Hipótesis nula.....	32
2.4 Sistema de variables .....	32
2.4.1 Variable independiente.....	32
2.4.2 Variable dependiente .....	32
2.4.3 Variables intervinientes .....	32
3. CAPITULO 3.....	33
3.1 Herramientas .....	34
3.2 Cadena de Markov para el análisis de patrones .....	36
3.3 Sensores .....	37
3.4 Procesamiento de imagen .....	38
3.5 Extracción de puntos .....	40
3.6 Creación, entrenamiento y reconocimiento de las señas .....	41
3.7 Diagramas UML.....	45
3.7.1 Diagrama de clases.....	45
3.7.2 Diagrama de objetos .....	48
3.7.3 Diagrama de componentes .....	48
3.7.4 Diagrama de casos de Uso .....	50

3.7.5 Diagrama de secuencias .....	50
4. CAPITULO 4.....	54
4.1 Resultados .....	54
4.2 Conclusiones .....	60
4.3 Futuras líneas de investigación .....	61
ANEXOS .....	63
ANEXO A. Modelo archivo configuración. Formato XML .....	63
ANEXO B. Modelo para los patrones. Formato XML .....	64
ANEXO C. Especificación casos de uso .....	65
ANEXO D. Módulos correspondientes a las señas .....	75
ANEXO E. Formato de encuesta .....	77
BIBLIOGRAFIA .....	78



## TABLA DE FIGURAS

	Pág.
Figura 1. Dimensiones del lenguaje (Bloom and Lahey's model) .....	18
Figura 2. Clasificación de las vocales .....	19
Figura 3. Morfología del lenguaje.....	21
Figura 4. Sintaxis del lenguaje .....	21
Figura 5. Diccionario dactilológico colombiano .....	23
Figura 6. Referencia orientación y movimiento de la mano .....	24
Figura 7. Señal agradecimiento LSC .....	25
Figura 8. Partes del oído.....	26
Figura 9. Imagen RGB e imagen profundidad .....	38
Figura 10. Imagen profundidad con colores asignados .....	38
Figura 11. Sub-imagen profundidad mano derecha.....	39
Figura 12. Procesamiento de imagen .....	40
Figura 13. Articulaciones procesadas .....	40
Figura 14. Articulaciones de las manos .....	40
Figura 15. Diagrama de clases (Interface) .....	45
Figura 16. Diagrama de clases (Flujo de datos Kinect y filtrado de imagen) .....	46
Figura 17. Diagrama de clases (Componente ModeloMarkov y Excepciones) .....	47
Figura 18. Diagrama de objetos.....	48
Figura 19. Diagrama de componentes.....	49
Figura 20. Diagrama casos de uso .....	50
Figura 21. Diagrama de secuencia (Selección de módulo).....	51
Figura 22. Diagrama de secuencia (Señales dactilológicas) .....	52
Figura 23. Diagrama de secuencia (Señales no dactilológicas) .....	53

## TABLA DE TABLAS

	Pág.
Tabla 1. Fonemas consonánticos del español .....	20
Tabla 2. Estructura del lenguaje .....	22
Tabla 3. Locaciones espaciales del señante .....	24
Tabla 4. Grados de hipoacusia .....	27
Tabla 5. Comparativa dispositivos periféricos.....	35
Tabla 6. Asignación del color para los datos de profundidad.....	38
Tabla 7. Articulaciones capturadas .....	40

## TABLA DE GRAFICOS

	Pág.
Gráfico 1. Ciclos para el desarrollo del aplicativo .....	34
Gráfico 2. Tiempo promedio de carga con respecto a colecciones y estados de la cadena de Markov .....	42
Gráfico 3. Tiempos de carga para los entrenamientos por modulo.....	43
Gráfico 4. Porcentajes de exactitud por cada modulo.....	44
Gráfico 5. Los tiempos de carga son aceptables .....	55
Gráfico 6. Los tiempos para reconocer el usuario son aceptables.....	55
Gráfico 7. La forma en que se presentan las señas es la adecuada .....	56
Gráfico 8. La manera de interactuar es la adecuada .....	56
Gráfico 9. Manera de interactuar Vs sencillez en su uso .....	57
Gráfico 10. Reconoce de forma rápida la seña.....	57
Gráfico 11. Reconoce de forma precisa la seña .....	58
Gráfico 12. Fue sencillo realizar la seña .....	58
Gráfico 13. Reconocimiento preciso Vs sencillez en realizar la seña .....	59
Gráfico 14. Aplicativo como apoyo en la enseñanza LSC .....	59

## TABLA DE ANEXOS

	Pág.
ANEXO A. Modelo archivo configuración. Formato XML.....	63
ANEXO B. Modelo para los patrones. Formato XML.....	64
ANEXO C. Especificación casos de uso.....	65
ANEXO D. Módulos correspondientes a las señas.....	75
ANEXO E. Formato de encuesta .....	77

## GLOSARIO<sup>1</sup>

- **ALFABETO MANUAL O DACTILOLÓGICO:** Sistema de representación simbólica o icónica de las letras de los alfabetos de las lenguas orales-escritas por medio de las manos.
- **COMUNIDAD SORDA:** Es el grupo social de personas que se identifican a través de la vivencia de la sordera y el mantenimiento de ciertos valores e intereses comunes y se produce entre ellos un permanente proceso de intercambio mutuo y de solidaridad. Forman parte del patrimonio pluricultural de la Nación y que, en tal sentido, son equiparables a los pueblos y comunidades indígenas y deben poseer los derechos conducentes.
- **HIPOACUSIA:** Disminución de la capacidad auditiva de algunas personas, la que puede clasificarse en leve, mediana y profunda.
- **LENGUA DE SEÑAS:** Es la lengua natural de una comunidad de sordos, la cual forma parte de su patrimonio cultural y es tan rica y compleja en gramática y vocabulario como cualquier lengua oral. Se caracteriza por ser visual, gestual y espacial. Como cualquiera otra lengua tiene su propio vocabulario, expresiones idiomáticas, gramáticas, sintaxis diferentes del español. Los elementos de esta lengua (las señas individuales) son la configuración, la posición y la orientación de las manos en relación con el cuerpo y con el individuo, la lengua también utiliza el espacio, dirección y velocidad de movimientos, así como la expresión facial para ayudar a transmitir el significado del mensaje, esta es una lengua viso-gestual.
- **SORDO:** Es todo aquel que no posee la audición suficiente y que en algunos casos no puede sostener una comunicación y socialización natural y fluida en lengua oral alguna, independientemente de cualquier evaluación audio-métrica que se le pueda practicar.
- **SORDO BILINGÜE:** Es todo aquel que vive una situación bilingüe en Lengua de Señas Colombiana y castellano escrito u oral según el caso, por lo cual utiliza dos (dos) lenguas para establecer comunicación tanto con la comunidad sorda que utiliza la Lengua de Señas, como con la comunidad oyente que usa castellano.
- **SORDO CIEGO:** Es aquella persona que en cualquier momento de la vida puede presentar una deficiencia auditiva y visual tal que le ocasiona serios problemas en

---

<sup>1</sup> Tomado de: INSOR, Glosario. Internet: ([insor.gov.co/atencion-al-ciudadano/glosario/](http://insor.gov.co/atencion-al-ciudadano/glosario/))

la comunicación, acceso a información, orientación y movilidad. Requiere de servicios especializados para su desarrollo e integración social.

- **SORDO HABLANTE:** Es todo aquel que adquirió una primera lengua oral. Esa persona sigue utilizando el español o la lengua nativa, puede presentar restricciones para comunicarse satisfactoriamente y puede hacer uso de ayudas auditivas.
- **SORDO SEÑANTE:** Es todo aquel cuya forma prioritaria de comunicación e identidad social se define en torno al uso de Lengua de Señas Colombiana y de los valores comunitarios y culturales de la comunidad de sordos.

## **Aplicativo para apoyar el proceso de aprendizaje del lenguaje de señas hacia un oyente mediante Microsoft Kinect®**

Durante muchos años la especie humana ha evolucionado desarrollando sus capacidades cognitivas y lingüísticas, la comunicación oral es uno de los principales atributos que la humanidad posee, con la que es posible la interacción con la sociedad, transmitiendo e intercambiando ideas con otro individuo. No obstante, no todos los seres humanos poseen la capacidad de comunicarse mediante el lenguaje hablado, un ejemplo de ello son las personas con discapacidad auditiva, que a causa de su imposibilidad no logran adquirir el dominio de la comunicación lingüística, debido a que su interacción con el léxico y la fonética de la lengua española fue mínima o nula.

Por tal motivo el lenguaje o lengua de señas es el método alternativo con el cual personas con discapacidad auditiva pueden comunicar, expresar e intercambiar ideas, mediante movimientos manuales, expresiones faciales, gestos y movimientos corporales. Con dicho sistema de expresión gestual, se establece un medio de comunicación entre una persona sorda y alguien que domina el lenguaje de señas, recibiendo y transmitiendo información, pero al momento de crear ese mismo canal de comunicación entre un sordo y un oyente, este canal es bloqueado debido a que aquello que se trata de manifestar o comunicar por medio de la lengua de señas, es mal interpretado por el oyente, creando un impedimento para mantener una relación de comunicación.

Dicha brecha en la comunicación colabora a que aumente la exclusión social hacia la comunidad sorda, quitándoles oportunidades sociales, educativas y laborales, por este motivo el presente proyecto plantea el desarrollo de un software que con ayuda de la cámara RGB, el sensor de profundidad y el reconocimiento corporal de Kinect, en conjunto con el procesamiento de señales e inteligencia artificial le otorgue a las personas oyentes elementos básicos de la lengua de señas Colombia, con el fin de apoyarlas en el proceso de aprendizaje de la lengua de señas, mejorando sus conocimientos y habilidades sobre la misma, y de esa manera al momento de crear un canal comunicativo con personas con discapacidad auditiva, la brecha existente en la comunicación sea menor, facilitando no solo la inclusión social de este grupo minoritario, sino también que personas oyentes sean elementos activo e involucrado dentro de la comunidad sorda.

## **1. CAPITULO 1**

### **1.1 Alcance**

Con ayuda de Kinect en su primera versión y el reconocimiento de patrones con modelos de inteligencia artificial y procesamiento de señales, se realizara un aplicativo con contenidos de imagen, video e interactividades, que proporcionen al usuario diferentes elementos de la lengua de señas Colombia como lo es el diccionario dactilológico, los números, los colores, entre otros. Brindándole al oyente un apoyo en el aprendizaje de esta lengua, con el fin de facilitar y mejorar la interpretación, y el entendimiento de los gestos del lenguaje natural de la comunidad sorda colombiana.

### **1.2 Limites**

La primera generación de Kinect posee dos cámaras, una de color (RGB) y la otra de profundidad<sup>2</sup>, las cuales en su interacción conjunta reconocen una figura humana. Dicha figura está compuesta por veinte articulaciones<sup>3</sup>, este esqueleto reconocido por Kinect no posee la capacidad de captar movimientos con las manos, específicamente con los dedos (HandTracking), es por este motivo que el reconocimiento de los elementos de las señas dactilológicas se realizara por medio de capturas de imágenes y filtrados de la misma.

La distancia a la cual se posicionara la persona oyente del Kinect, debe estar dentro de los límites donde las cámaras puedan capturar los movimientos, este campo de visión va desde 1.2 metros hasta 3.5 metros<sup>4</sup>, esto con el fin de realizar un reconocimiento más preciso.

Debido a la distancia entre el dispositivo y el usuario, no existe un margen aceptable para realizar un seguimiento y reconocimiento de gestos faciales, (FaceTracking), es por este motivo que las señas ingresadas dentro del sistema no corresponden a este tipo de señas.

Con el presente trabajo no se espera llegar a establecer un traductor de la lengua de señas Colombia hacia la lengua Castellana, sin embargo el aplicativo será capaz de reconocer de manera individual diferentes señas y configuraciones manuales.

---

<sup>2</sup>MICROSOFT, Kinect for Windows Sensor Components and Specifications. Internet: (<http://msdn.microsoft.com/en-us/library/jj131033.aspx>.)

<sup>3</sup>MICROSOFT, Skeletal Tracking. Internet: <http://msdn.microsoft.com/en-us/library/hh973074.aspx>

<sup>4</sup>Ibid.



### **1.3 Objetivos**

#### **1.3.1 General**

- Ofrecer un aplicativo que con contenido multimedia<sup>5</sup> sirva de apoyo para el aprendizaje del lenguaje de señas a personas oyentes, de forma sencilla y didáctica.

#### **1.3.2 Específicos**

- Reconocer e identificar cada uno de los patrones asociados al diccionario dactilológico colombiano, mediante la tecnología Kinect en su primera versión con el SDK 1.8.
- Identificar el componente morfológico de la seña<sup>6</sup>, mediante el reconocimiento de gestos dinámicos, otorgando ejemplos prácticos para el uso de la seña.

---

<sup>5</sup> Entiéndase como multimedia contenidos de audio, video, animaciones e interactividades.

<sup>6</sup> Entiéndase como seña unidad mínima lingüística con significado por si misma - palabra

## 2. CAPITULO 2

### 2.1 Revisión bibliográfica

Al exponer el término lenguaje, nos referimos a todo el conjunto de signos y sonidos que utiliza un individuo, para comunicarse con otro individuo de su misma especie, por los cuales expresa lo que piensa o siente acerca de una cuestión determinada<sup>7</sup>, de manera similar Sapir (1921) lo define como un método exclusivamente humano, y no instintivo, de comunicar ideas, emociones y deseos por medio de un sistema simbólico, producidos de manera deliberada.<sup>8</sup>

De esta manera podemos entender al lenguaje como el recurso, que hace posible la comunicación entre individuos, para el caso particular de los humanos a diferencia de otras especies, esta herramienta se encuentra altamente desarrollada y estructurada, ya sea en un lenguaje coloquial, natural o técnico.

Para dicha estructura del lenguaje, el modelo desarrollado por Bloom y Lahey (1978)<sup>9</sup> es útil para mostrar cómo se interrelacionan las habilidades lingüísticas (Figura 1).

Figura 1. Dimensiones del lenguaje (Bloom and Lahey's model)



Tomado de: <https://www.slc.cambridgeshire.nhs.uk>

<sup>7</sup>Subgerencia Cultural del Banco de la República. "Lengua y lenguaje". Internet: ([http://www.banrepcultural.org/blaavirtual/ayudadetareas/comunicacion/lengua\\_y\\_lenguaje](http://www.banrepcultural.org/blaavirtual/ayudadetareas/comunicacion/lengua_y_lenguaje)), 2015.

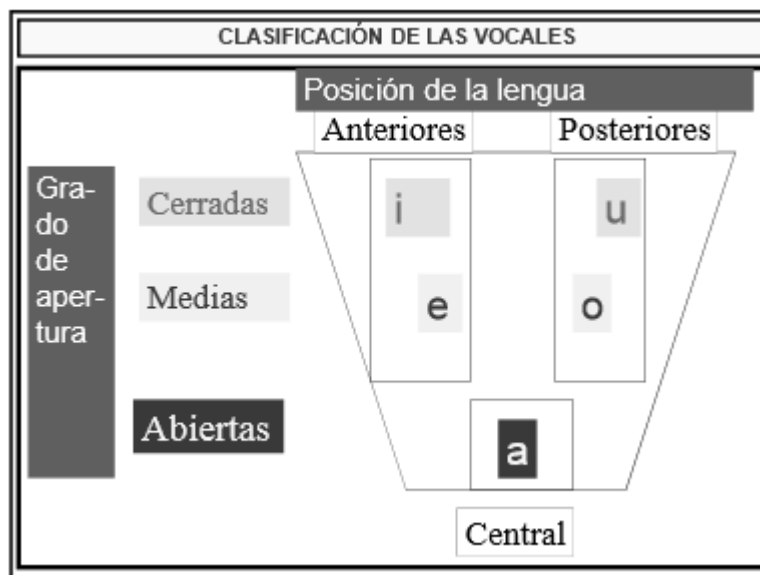
<sup>8</sup>SAPIR, Edward, El lenguaje: Introducción al estudio del habla. 1921, p. 8-10.

<sup>9</sup>BLOOM, L & LAHEY, M. Language development and language disorders. New York, Wiley, 1978

En dicho modelo encontramos 3 diferentes dimensiones que conforman el lenguaje, la dimensión 'Forma', la cual abarca todo sistema fonológico, el cual describe y analiza la naturaleza física de la producción de los sonidos, desde el aspecto articulatorio, de los aparatos responsables del habla, como lo son el respiratorio y el bucofonatorio<sup>10</sup>. El elemento de estudio son los fonemas, los cuales son las unidades más básicas del lenguaje, que son las vocales y las consonantes, las cuales son descritas, desde el punto de vista alófono (sonidos).

(Ejemplo: casa -> /k/ /a/ /s/ /a/; contiene 4 fonemas)

Figura 2. Clasificación de las vocales



[http://www.uam.es/personal\\_pdi/psicologia/agonzale/LgTema1.pdf](http://www.uam.es/personal_pdi/psicologia/agonzale/LgTema1.pdf)

<sup>10</sup> Fonética. Internet: ([www.salonhogar.com/espanol/lenguaje/lengua/fonetica.htm](http://www.salonhogar.com/espanol/lenguaje/lengua/fonetica.htm))

Tabla 1. Fonemas consonánticos del español

FONEMAS CONSONÁNTICOS DEL ESPAÑOL														
	Bilabial		Labiodental		Dental		Interdental		Alveolar		Palatal		Velar	
	sor.	son.	sor.	son.	sor.	son.	sor.	son.	sor.	son.	sor.	son.	sor.	son.
Oclusiva	p	b			t	d							k	g
Fricativa			f				θ		s			ʃ	x	
Africada											tʃ			
Nasal		m								n		ɲ		
Lateral										l		ʎ		
Vibrante simple										r				
Vibrante múltiple										rr				

[http://www.uam.es/personal\\_pdi/psicologia/agonzale/LgTema1.pdf](http://www.uam.es/personal_pdi/psicologia/agonzale/LgTema1.pdf)

De igual manera la dimensión 'Forma', contempla el aspecto morfosintáctico del lenguaje, de la cual se derivan la morfología y la sintaxis, la primera de ellas se encarga de explicar la estructura interna de las palabras y el proceso de formación de palabras mediante dos estructuras básicas, '*lexema*' comúnmente llamado raíz o parte invariable y '*morfema*'<sup>11</sup> o parte variante de la palabra, para realizar la respectiva categorización de las unidades gramaticales.<sup>12</sup> (Figura 3)

(Ejemplo: Niñas, Niños, Niñito -> lexema [Niñ], morfema [as, os, ito])

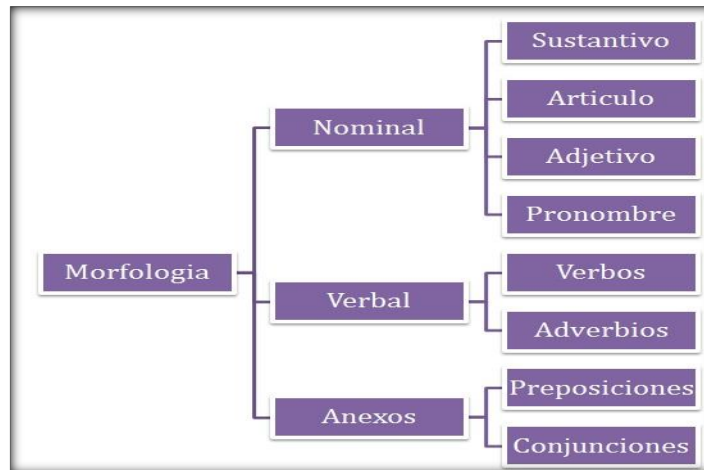
Mientras que la sintaxis describe la combinación de los anteriores valores morfológicos para formar desde estas unidades mínimas como una palabra, a unidades superiores como la oración, pasando por unidades de organización gramatical intermedias como el sintagma (Figura 4).<sup>13</sup>

<sup>11</sup> La Morfología lingüística en Gramática. Internet: (<http://www.deperu.com/abc/gramatica/2196/la-morfologia-linguistica-en-gramatica>)

<sup>12</sup> O'SHANAHAN, Isabel, Enseñanza del lenguaje oral y las teorías implícitas del profesorado. Trabajo de grado. Universidad de la Laguna. Humanidades y ciencias sociales. 1996. p. 66

<sup>13</sup> Ibid., p. 55-56.

Figura 3. Morfología del lenguaje



Tomado de: <http://ladosalvaje.files.wordpress.com/2011/11/morfologia.jpg>

Figura 4. Sintaxis del lenguaje



Tomado de: <http://ladosalvaje.files.wordpress.com/2011/11/sintaxis.jpg>

La segunda dimensión del modelo planteado por Bloom y Lahey es el ‘Contenido’, el cual describe toda la semántica lingüística, es decir la interpretación del significado del signo lingüístico, mediante tres componentes básicas, el significante (parte material del signo), el significado (imagen mental del signo) y el referente (elemento real del signo)<sup>14</sup>. Los significados dependen del rasgo conceptual del signo, de manera objetiva o denotación (común entre hablantes), o manera subjetivo o connotación (común entre ciertos hablantes).<sup>15</sup>

La ultima dimensión el ‘uso’, que especifica la pragmática del lenguaje, que se refiere al diferente uso del lenguaje en la comunicación, es decir, las condiciones que determinan las diferentes formas de emplear un mismo enunciado por parte

<sup>14</sup> Semántica. Internet: ([www.profesorenlinea.cl/castellano/Semantica1.htm](http://www.profesorenlinea.cl/castellano/Semantica1.htm))

<sup>15</sup> Ibid.

de un hablante en diferentes contextos, de igual manera con su respectiva interpretación por parte del destinatario.<sup>16</sup>

Tabla 2. Estructura del lenguaje

	Esta noche  ladraron  los perros		
<b>Fonético</b>	/esta'notše la'draron los'perros/		
<b>Semántico</b>	locativo	acción	agente
<b>Sintáctico</b>	circunstan.	verbo	sujeto
<b>Pragmático</b>	tema	rema	

Tomado de: [http://www.sld.cu/galerias/pdf/sitios/rehabilitacion-logo/introducción\\_a\\_la\\_psicolinguistica.pdf](http://www.sld.cu/galerias/pdf/sitios/rehabilitacion-logo/introducción_a_la_psicolinguistica.pdf)

Se debe tener en cuenta que también existen otros diferentes tipos de lenguajes, como sería el caso, por ejemplo, del lenguaje de señas, también llamado lengua de señas, el cual es un lenguaje natural de contenidos gestuales, espaciales, faciales y visuales, como el medio de comunicación, para qué personas con discapacidad auditiva, pueda establecer una relación comunicativa con su entorno social. A pesar de que el lenguaje de señas es utilizado en todo el mundo, no es un lenguaje universal, puesto que existen cientos de lenguas de señas, por ejemplo, Lengua de Señas Japón (o Nihon Shuwa, JSL), Lengua de Señas Británica (BSL) o Lengua de Señas España (LSE) entre muchas otras.<sup>17</sup>

No obstante existen diversas familias lingüísticas las cuales agrupan en su gran mayoría las leguas de señas del mundo, de las cuales encontramos:<sup>18</sup>

- Lengua de Señas Británica, Australia y Nueva Zelanda (BANZSL)
- Lengua de Señas Francia (LSF)
- Lengua de Señas Alemania (GSL)
- Lengua de Señas Japón (JSL)
- Lengua de Señas Suecia (SSL)

<sup>16</sup> Pragmática. Internet: ([www.wdb.ugr.es/~auladeletras/?page\\_id=13](http://www.wdb.ugr.es/~auladeletras/?page_id=13))

<sup>17</sup> World Federation of the Deaf (WFD), Sign Language. Internet: (<http://wfdeaf.org/our-work/focus-areas/sign-language>)

<sup>18</sup> Deafness Cognition and Language Research Centre, Is sign language the same the world over?, Internet: (<http://www.ucl.ac.uk/dcal/faqs/questions/bsl/question6>)

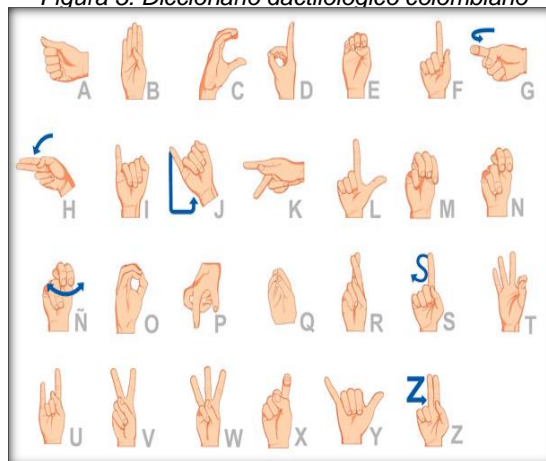
Para el caso particular de Colombia, su sistema de señas ‘*Lengua de Señas Colombia (LSC)*’, fue influenciado en los años 50 por la Lengua de Señas Española (LSE)<sup>19</sup>, y en los años 70 por la Lengua de Señas Americana (ASL)<sup>20</sup>, que a su vez derivan de la Familia de Lengua de Señas Francia (LSF).

Para el año de 1996 la LSC, fue reconocida oficialmente, durante del gobierno del expresidente Ernesto Samper Pizano, mediante la ley 324 del 96, en su artículo 2° que expresa:

*“El Estado colombiano reconoce la lengua de señas como propia de la comunidad sorda del país”<sup>21</sup>*

Dicha Lengua de señas posee su propio diccionario dactilológico que deriva del alfabeto manual internacional (LSE), el cual representa de forma manual las 27 letras del alfabeto de la lengua oral-escrita del país. (Figura 5)

Figura 5. Diccionario dactilológico colombiano



Fuente: <http://www.makaia.org/recursos.shtml?apc=h1d1--->

El LSC al ser una lengua natural, posee una estructura de forma, para ello debemos entender el componente espacial, el cual describe las locaciones

<sup>19</sup> INSOR. Diccionario básico de la lengua de señas colombiana. Bogotá: 2006.

<sup>20</sup> OVIEDO, Alejandro. Apuntes para una gramática de la lengua de señas colombiana. Cali: Universidad del Valle-INSOR. 2000. p. 5.

<sup>21</sup> COLOMBIA. CONGRESO DE LA REPUBLICA. Ley 324. (11, octubre, 1996). Por la cual se crean algunas normas a favor de la Población Sorda. Diario oficial. Bogotá, D.C., 1996. no. 42899.

definidas por coordenadas en el espacio de las señas con respecto al señante<sup>22</sup>. Dichas locaciones se dividen en 210, y generan las diferentes posibilidades tridimensionales de segmentación (Tabla 3)<sup>23</sup>

*Tabla 3. Locaciones espaciales del señante*

A lo ancho	A lo largo	A lo alto
V <sub>3</sub> derecha	Próximo	Coronilla
V <sub>2</sub> derecha	Medio	Frente
V <sub>1</sub> derecha	Distante	Nariz
V <sub>0</sub>		Boca
V <sub>1</sub> izquierda		Barbilla
V <sub>2</sub> izquierda		Cuello
V <sub>3</sub> izquierda		Hombro
		Pecho
		Costillas
		Abdomen

Tomado de: Aspectos relevantes del discurso en Lengua de Señas Colombiana

Las diferentes locaciones componen lo que el lingüista venezolano Alejandro Oviedo, en el libro “Apuntes para una gramática de la Lengua de Señas Colombiana”, denomina matriz articulatoria, describe la postura de la mano, cada una de sus partes móviles, su respectiva orientación y ubicación espacial, denominada configuración manual (CM).

*Figura 6. Referencia orientación y movimiento de la mano*



Tomado de: Diccionario básico de la lengua de señas colombiana

<sup>22</sup>BARRETO, Alex G y CORTÉS, Yenny M. Aspectos relevantes del discurso en Lengua de Señas Colombiana (LSC). Bogotá: Universidad Nacional de Colombia. P. 252.

<sup>23</sup> Ibid. P 253.



La segunda matriz nombrada por Oviedo, la matriz segmental, hace referencia a los diferentes segmentos que conforman una determinada seña, analizando cada característica de la acción realizada<sup>24</sup>, en esta matriz se identifican tres tipos de segmentos: de movimiento, de detención y de transición. Que determinan el cambio producido en la configuración manual, la cual reconoce si dentro de una específica seña existe o se producen movimientos dentro de la matriz espacial, así mismo identifica detenciones o diferentes transiciones de la mano o las manos durante el desarrollo de la seña.

Con ello se define la estructura de la seña, de manera semejante a la composición de sílabas en una palabra, las señas tienen un símil de las sílabas, los momentos, los cuales describen la composición de la seña. Ejemplo de ello la palabra 'gracias', en el idioma castellano está compuesta por dos sílabas 'gra'+ 'cias', de igual misma manera la seña 'gracias' (Figura 7), posee tres momentos, la mano en forma de palma situada en el mentón, seguida de un movimiento de la mano hacia adelante a una distancia media, finalmente terminada por una detención con la otra mano a la altura del pecho.<sup>25</sup>

Figura 7. Seña agradecimiento LSC



Tomado de: <http://www.ucn.edu.co/e-discapacidad/Documents/36317784-Diccionario-lengua-de-senas.pdf>

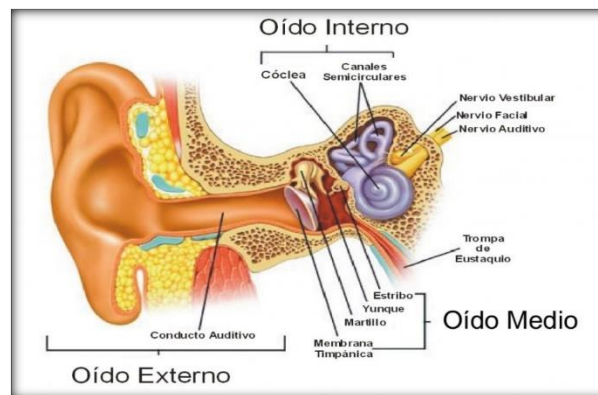
<sup>24</sup> AYALA C, Jaime A, Propuesta conceptual para la elaboración de un currículo para la lengua de señas colombiana a partir de lineamientos curriculares. En: Horizontes pedagógicos. Enero-Diciembre, 2010, vol. 12.

<sup>25</sup> INSOR. Diccionario básico de la lengua de señas colombiana. Bogotá: 2006. P. 485.

El tercer componente, la matriz de gestos no manuales, la cual contempla toda la información de carácter gestual, que no se realiza con las manos sino que se hace con los ojos, la boca, la nariz, las mejillas, las cejas, la cabeza y el cuerpo, los cuales refuerzan, complementan y dan sentido sintáctico a la seña, en las matrices anteriormente mencionadas.

El LSC es utilizado en su gran mayoría, por personas colombianas con problemas auditivos, que afectan directamente al oído, el cual es el órgano responsable de la audición y el centro de equilibrio. El oído está compuesto por tres zonas: externa, media e interna (Figura 8)

Figura 8. Partes del oído



Tomado de: <http://tinnipad.com/es/i-rumori-nellorecchio/>

El oído externo está formado el pabellón auricular (oreja) y el conducto auditivo externo de 3 cm de longitud y 0.6 cm de diámetro aproximadamente<sup>26</sup>, por donde pasan las ondas sonoras hacia el oído medio, el cual es una cámara de aire ubicada entre el oído externo y el oído interno, que contiene la membrana timpánica, la cual es estimulada por las ondas sonoras, haciéndola vibrar. Estas series de vibraciones, se comunican al oído interno a través de los huesos móviles martillo, yunque y estribo, que conectan acústicamente el tímpano con el oído interno<sup>27</sup>, que contiene a su vez la cóclea, el vestíbulo y los canales semicirculares, que con ayuda de las células pilosas transmiten señales directamente al nervio auditivo<sup>28</sup>, y posteriormente enviadas al cerebro.

<sup>26</sup> Dr. Matamala, Apuntes Neuroanatomía- UFRO: Oído-externo. Universidad de la frontera. Temuco, Chile.

<sup>27</sup> Ibíd. Oído-externo.

<sup>28</sup> Ibíd. Oído-Interno.

Los rangos de audición con los cuales una persona debería escuchar se encuentran entre 0 y 20 dB<sup>29</sup>, sin embargo el la capacidad auditiva puede llegar hasta los 85 dB<sup>30</sup>, más de esa cifra se expone al oído a afecciones leves-graves y superando los 140 dB o umbral del dolor, se expone a daños graves-permanentes.<sup>31</sup>

Dentro de estas lesiones al oído, encontramos 3 tipos de afecciones: conductiva, perceptiva y mixta. La sordera conductiva se produce por obstrucciones al oído externo, como exceso de cerumen u otitis, o enfermedades que afectan levemente al tímpano o los huesos el oído, dicho tipo de sordera o hipoacusia es tratable ya sea médica o quirúrgicamente<sup>32</sup>. La sordera perceptiva o neurosensorial es causada por enfermedades que afectan a las células pilosas o a las fibras nerviosas en el oído interno, también puede ser causada de forma congénita<sup>33</sup>, este tipo de hipoacusia es de carácter grave y puede ser manejada con amplificadores de sonidos, implantes cocleares y audífonos. La sordera mixta es el resultado de la combinación de una sordera conductiva y una sordera perceptiva.<sup>34</sup>

De igual manera podemos clasificar la sordera o hipoacusia, según su grado de severidad.

*Tabla 4. Grados de hipoacusia*

Pérdida leve	20 a 40 dB
Pérdida moderada	40 a 60 dB
Pérdida severa	60 a 80 dB
Pérdida profunda	80 a 100 dB
Restos auditivos	más de 100 dB
Anacusia o Cofosis	No registra audición

Tomado de: <http://auditivos.lux.mx/index.php/perdida-auditiva/rangos-de-audicion>

La pérdida de audición leve y moderada (20-60 dB) se presentan algunos problemas para escuchar a cierta distancia y en un volumen moderado, la pérdida de audición severa (60-80 dB) conlleva problemas para captar sonidos sin ruido de

<sup>29</sup> Rangos de audición. Internet: ([auditivos.lux.mx/index.php/perdida-auditiva/rangos-de-audicion](http://auditivos.lux.mx/index.php/perdida-auditiva/rangos-de-audicion))

<sup>30</sup> Ibid.

<sup>31</sup> Instituto Nacional de la Sordera y otros Trastornos de la Comunicación, La pérdida de audición ocasionada por el ruido, NIH Pub. No. 09-4233, Actualizado en junio del 2009

<sup>32</sup> Dr. LAFUENTE, Antonio D., Tipos y causas de hipoacusia. Internet: (<http://sorderayvertigo.com/hipoacusias2>)

<sup>33</sup> Sordera neurosensorial, Internet: ([www.nlm.nih.gov/medlineplus/spanish/ency/article/003291.htm](http://www.nlm.nih.gov/medlineplus/spanish/ency/article/003291.htm))

<sup>34</sup> Los diferentes tipos de sordera. Internet: ([www.neurelec.com/es/sordera/sordera](http://www.neurelec.com/es/sordera/sordera))

fondo, para la sordera profunda (80-100 dB) se presentan dificultades para escuchar siempre; sólo escucha a una persona que habla alto y cerca, Los restos auditivos (más de 100 dB) a pesar que captar algún sonido es casi imposible, hay sonidos que logra escuchar, que sean de un tono muy elevado y la anacusia o cofosis, significa la pérdida de audición total.<sup>35</sup>

Todas estos tipos de afecciones al oído las padecen la comunidad sorda colombiana que según las cifras registradas por el DANE en el censo de 2005, el 6.47% de la población posee alguna limitación física, de las cuales el 17,3% posee limitaciones para oír aun con aparatos especiales y el 2,8% tiene algún limitante en el habla<sup>36</sup>, esto se reduce, que aproximadamente el 1% de la población en Colombia, pertenece a la comunidad sorda colombiana.

Dicha comunidad es apoyada por organismos sin ánimo de lucro, de los cuales los se destacan, FENASCOL, que tiene como misión principal ayudar a mejorar la calidad de vida de las personas con discapacidad auditiva, defendiendo sus derechos, realizando programas que respondan a sus necesidades<sup>37</sup>. De igual manera la entidad INSOR que con ayuda del ministerio de educación, promueven el desarrollo de la inclusión social a la comunidad sorda, desde el sector educativo<sup>38</sup>, estas y muchas entidades más están comprometidas con la población sorda colombiana en pro de su integración social.

De igual manera el presente trabajo está enfocado en la forma de mejorar el canal comunicativo entre personas con discapacidad auditiva y personas oyentes, facilitando la inclusión a la sociedad, por medio de un software que colaborara a personas oyentes en el proceso del aprendizaje del LSC, este software funcionara con ayuda del Kinect 1.0. Este hardware de Microsoft es un accesorio de la consola de videojuegos Xbox 360, que posee la característica de interactuar con el dispositivo mediante la interface natural de usuario (NUI), que con el reconocimiento de movimientos corporales o de voz, se pueda interactuar con la Xbox o una computadora.<sup>39</sup>

---

<sup>35</sup> DE LOS ANDES, Joaquin, Breve reseña sobre aspectos fundamentales de la sordera, Corporación de desarrollo de la comunicación. P 2-5.

<sup>36</sup> FENASCOL.Documento publicado. Acercamiento a la comunidad sorda su lengua y su cultura. P. 19 2013.

<sup>37</sup> Ibid. P 22.

<sup>38</sup> INSOR. Mision y vision. Internet: ([www.insor.gov.co/mision-y-vision](http://www.insor.gov.co/mision-y-vision))

<sup>39</sup> MICROSOFT. Primeros pasos con Kinect. Internet :([www.xbox.com/es-CO/Kinect/GetStarted](http://www.xbox.com/es-CO/Kinect/GetStarted))

## 2.2 Marco referencial

Así mismo, a día de hoy existen diversos trabajos que conceptualmente siguen el mismo enfoque que el presente proyecto, que de igual manera hacen uso de herramientas tecnológicas y sistemas informáticos para realizar reconocimiento y síntesis visuales de la lengua de señas, entre ellos encontramos el proyecto realizado por Michael Shantz y Howard Poizner en el año 1982, "*A computer program to synthesize American Sign Language*"<sup>40</sup>, donde sintetizan las secuencias de posiciones de las articulaciones de los brazos y manos correspondientes a una seña ASL, mediante imágenes, creando una representación en un plano tridimensional que varía en el tiempo, de esta manera con ayuda de operaciones matriciales otorgar la respectiva aproximación de la seña. La principal barrera durante su investigación corresponde al recurso tecnológico de la época, tanto de hardware como de software, que afectaron directamente a la eficiencia y procesamiento de la misma.

De igual manera encontramos referente el libro recopilatorio "*Gesture-based communication in human computer interaction*"<sup>41</sup>, que describe todos avances y proyectos relacionados al International Gesture Workshop, (GW'99), celebrado en Francia en el año 1999, donde describen las nuevas tecnologías referentes a la percepción computarizada y la producción de gestos, mediante la localización, la segmentación de imágenes y el reconocimiento del lenguaje de signos, donde para la síntesis de las señas los datos obtenidos son modelados en segmentos poligonales conectados entre sí, para realizar la extracción de los valores de apertura, de las diferentes articulaciones del esqueleto, que en conjunto con las coordenadas espaciales, métodos matemáticos y estadísticos realizan aproximaciones al reconocimiento de los patrones asociados a las configuraciones manuales ASL.

Otro trabajo que cabe resaltar es el realizado en el año 2005 por Nguyen Dang Binh, Enokida Shuichi y Toshiaki Ejima, "*Real-Time Hand Tracking and Gesture Recognition System*"<sup>42</sup>, en donde hacen uso del reconocimiento de color (ColorTracking) mediante el filtro Kalman, para identificar regiones en la imagen correspondientes a el color de la piel humana, y de esta manera extraer la figura más parecida a una mano, para aplicarle procesos de binarizado fijando un valor umbral, en este punto por medio de un algoritmo HandTracking, se calcula la posición de la mano en cuestión, otorgándole datos de coordenadas espaciales a un pseudo-modelo oculto de Markov bidimensional, que realiza los procesos de

---

<sup>40</sup> SHANTZ, Michael y POIZNER, Howard: A computer program to synthesize American Sign Language. En: Behavior Research Methods & Instrumentation. 1982, vol. 14(5), p. 467-474.

<sup>41</sup> LOSSON, Oliver y VANNOBEL, Jean-Marc: Sign Specification and Synthesis. En: Gesture-Based Communication in Human-Computer Interaction: International Gesture Workshop, GW'99. Marzo de 1999, Francia, p 139-250.

<sup>42</sup> DANG BINH, Nguyen, SHUICHI, Enokida y EJIMA, Toshiaki: Real-Time Hand Tracking and Gesture Recognition System. En: GVIP 05 Conference (19-21, Diciembre: El Cairo, Egipto). 2005.

reconocimiento de los elementos de ASL. Un punto clave para la correcta detención, procesamiento y reconocimiento de estas señas, corresponde al uso de mangas largas por parte del usuario de un color diferente al de la piel, para excluir completamente la mano del antebrazo debido a la detención de color, otro factor determinante son los valores umbrales que son asignados tanto en la detención de color, como en el filtro de binarizado, debido a que no todos los usuarios contarán con la misma tonalidad de piel y no siempre se dispondrá con las mismas condiciones de luz-contraste para binarizar la imagen de la mejor manera.

Un trabajo similar al que se realizó en 2005, fue propuesto por Enrique Antón López en su trabajo de grado *“Reconocimiento automático de lenguaje de signos: Lenguaje ASL”*<sup>43</sup> en el año 2009, que para superar los limitantes de los umbrales para los filtros de binarizado, no utiliza la segmentación de imagen en los canales RGB (Rojo, Azul, Verde), en cambio a ello utiliza la segmentación en los espacios HVS de la imagen (Tono, Brillo, Saturación), con lo cual la variable luminosidad, sale de la escena haciendo que la extracción de datos de la imagen sea más completa, aun así la variable contraste sigue en juego, por ello el espacio de reconocimiento debe tener colores totalmente contrastados al color de la piel del usuario.

En año 2012, Esteban Chacón, Darwin Aguilar y Fabián Sáenz, implementan el proyecto *“Desarrollo de una Interfaz para el Reconocimiento Automático del Lenguaje de Signos”*<sup>44</sup>, donde dan solución el reconocimiento de señas, con ayuda del componente tecnológico AcceleGlove, que fue creado por Jack Maxwell fundador de AnthroTronix, que consiste en un guante con seis diferentes acelerómetros de tres ejes, que recopila datos en tiempo real de la posición de la mano y los dedos en un espacio tridimensional, todos estos datos son evaluados y comparados con los modelos previamente establecidos, para decidir la mejor coincidencia y otorgar el reconocimiento de la seña, las cuales corresponden a señas dactilológicas y señas que exclusivamente son realizadas con una mano.

Otro proyecto con mayor similitud al presente trabajo fue realizado por Daniel Martínez Capilla en el año 2012 *“Traductor de Lenguaje de Signos utilizando Microsoft Kinect XBOX 360”*<sup>45</sup>, en donde con ayuda del Kinect realiza la extracción

---

<sup>43</sup> LÓPEZ, Enrique Antón. Reconocimiento automático de lenguaje de signos: Lenguaje ASL. Trabajo grado Ingeniería técnica en informática de sistemas. Barcelona: Universidad de Barcelona. Facultad de Matemáticas, 2009.

<sup>44</sup> CHACÓN, Esteban, AGUILAR, Darwin y SÁENZ Fabián. Desarrollo de una Interfaz para el Reconocimiento Automático del Lenguaje de Signos. Trabajo de grado. Valle de los Chillos, Ecuador: Universidad De Las Fuerzas Armadas – ESPE. Departamento de Eléctrica y Electrónica en Redes y Comunicación de Datos. 2013.

<sup>45</sup> CAPILLA MARTINEZ, Daniel. Sign language translator using Microsoft Kinect XBOX 360”. Master tesis ERASMUS MUNDUS Masters Vibot. Knoxville, USA. Departamento de ingeniería electrónica y ciencias de la computación. 2012.

de la figura del usuario, tomando como referencia las articulaciones correspondientes al torso, esta secuencia de datos es extraída y procesada en un Nearest Group Dynamic Time Warping (NG-DTW), que es un algoritmo que realiza comparaciones entre dos diferentes secuencias de datos buscando similitudes. De esta manera, se compara la seña a reconocer contra las señas que están en el diccionario, buscando la mejor coincidencia, y así obtener el reconocimiento de la seña; con esta metodología gana el reconocimiento para señas con una mayor velocidad, aunque pierde el reconocimiento del diccionario dactilológico, además de esto el sistema está compuesto por únicamente 14 diferentes señas, no oficiales.

El presente proyecto busca que con ayuda de Kinect, con su cámara, sensores y funcionalidad de SkeletonTracking, ofrecer el reconocimiento de diferentes señas LSC con un total de 120, incluyendo el diccionario dactilológico que será procesado desde la imagen de profundidad, esto con el fin de eliminar las variables de luz-contraste que en otros trabajos fueron inconvenientes, de igual manera partiendo de este tipo de imagen se elimina el uso de una escena contrastada con respecto al color de piel del usuario, ganando que cualquier tonalidad de piel es reconocida por el sensor de profundidad.

Así mismo las señas integradas en el aplicativo corresponden, tanto a señas dinámicas y estáticas a una o dos manos, que con ayuda de un Modelo Oculto de Markov previamente entrenado con una sucesión de señas, hará el respectivo reconocimiento.

## **2.3 Sistema de hipótesis**

### **2.3.1 Hipótesis de trabajo**

El desarrollo del aplicativo ofrecerá apoyo en el proceso de aprendizaje de la Lengua de Señas Colombia para una persona oyente, que por medio de ayudas multimedia como videos, interactividades y prácticas, facilitando la comprensión y entendimiento de dicho lenguaje.

### **2.3.2 Hipótesis nula**

Un aplicativo que ofrece un apoyo por medio de ayudas multimedia a las personas oyentes en el proceso de aprendizaje de la Lengua de Señas Colombia, no facilitara ni ayudara en el aprendizaje, la comprensión y entendimiento de dicho lenguaje.

## **2.4 Sistema de variables**

### **2.4.1 Variable independiente**

- Aplicativo de apoyo en el proceso de aprendizaje de la Lengua de Señas Colombia.
- Señas del diccionario de la lengua de señas Colombia.

### **2.4.2 Variable dependiente**

- Patrones reconocidos por Kinect asociados a las señas.
- Reconocimiento del esqueleto del Kinect
- Secuencia de puntos extraídos de las señas
- Procesamiento de las imágenes

### **2.4.3 Variables intervinientes**

- Métodos didácticos de enseñanza-aprendizaje.
- Personas que utilizaran el aplicativo.
- Kinect en su primera versión.
- SDK Kinect V1.8.
- Tiempos de carga en entrenamiento y reconocimiento de la cadena de Markov
- Cantidad de puntos para cada una de las señas
- Valores de convergencia y probabilidad de transición en cada proceso de aprendizaje de la cadena



### 3. CAPITULO 3

Para la implementación del sistema propuesto, se utilizó la metodología de desarrollo corresponde al método iterativo y creciente o modelo incremental, esto debido a que el prototipo inicial se sub-dividió en diferentes etapas o fases, cada una dando un valor específico al producto final, teniendo como particularidad que individualmente cada etapa del proceso ofrece una funcionalidad específica y funciona de manera individual.

Para cada iteración realizada se efectuaron los diferentes 4 pasos correspondientes a Análisis, Diseño, Implementación y Pruebas, en la etapa de Análisis se evalúa la funcionalidad que ofrece la correspondiente iteración dentro del producto final, teniendo en cuenta cada uno de los riesgos y limitaciones tanto de componentes de hardware como de software que puedan afectar el ciclo. Enseguida se realiza la etapa de diseño que corresponde a plasmar, con ayuda de diagramas UML la estructura de los componentes del presente ciclo, una vez terminado se inicia con la etapa de implementación o codificación, según lo definido en los diagramas y las funcionalidades analizadas, en el caso particular toda la implementación esta codificada bajo el lenguaje de programación C#.

Una vez finalizado la codificación se evalúa el resultado obtenido de la etapa, verificando si cumple con el objetivo propuesto y la funcionalidad especificada, de ser así se avanza a la nueva iteración, en caso contrario se reevalúa la etapa teniendo en cuenta los conflictos hallados.

El aplicativo se dividió en 9 diferentes etapas (Grafico 1), cada una correspondiente a un ciclo completamente funcional, los cuales corresponden a:

- **Interface:** Integración de los componentes visuales (MahApp.Metro y WpfAnimateGif) a aplicativo, que con ayuda de XAML se crean las pantallas.
- **Cámara color:** Integración y manipulación de los datos obtenidos por la cámara de color del Kinect, creando una imagen RGB.
- **Sensor profundidad:** Integración y manipulación los valores de profundidad dados por el sensor, recreando una imagen que contenga la figura de la mano.
- **Filtrado de Imagen:** Realizar diferentes segmentaciones y filtros de imagen para obtener el contorno de la mano-.
- **Seguimiento esqueleto:** Integración los datos dados por el Kinect (SkeletonTracking), para realizar el seguimiento y captura de los puntos de interés.
- **Extracción de puntos:** Recopilación y almacenamiento de los puntos que componen cada seña, creando el archivo de entrenamiento XML.

- **HMM:** Creación del modelo de Markov.
- **Entrenamiento cadena:** Agregar las diferentes observaciones para cada una de las señas las señas, dentro de la cadena mediante las observaciones del archivo de entrenamiento.
- **Reconocimiento patrón:** Comparación entre las observaciones en la cadena y la nueva observación, examinando similitudes para entregar la seña reconocida.

Gráfico 1. Ciclos para el desarrollo del aplicativo






Durante cada una de las iteraciones, todos los elementos generados, creados y modificados serán guardados en un repositorio de archivos, el Team Foundation Server bajo la dirección [www.robinsoncastro.visualstudio.com](http://www.robinsoncastro.visualstudio.com), que está directamente conectado con Visual Studio, en este repositorio estará almacenado todo el avance del proyecto.

### 3.1 Herramientas

Un componente fundamental para la correcta implementación del aplicativo propuesto, corresponde a la captura de imágenes, video y movimiento, por ello se evaluaron la especificaciones de los diferentes dispositivos periféricos (Hardware),

que ofrecen las características requeridas. Entre ellos encontramos Microsoft Kinect<sup>46</sup>, ASUS Xtion<sup>47</sup> y PrimeSense Carmine<sup>48</sup>.

*Tabla 5. Comparativa dispositivos periféricos*

	<b>Kinect</b> 	<b>ASUS Xtion</b> 	<b>PrimeSense Carmine</b> 
<b>Campo de visión</b>	57° H 43° V ± 27° Eje	58° H 45° V 70° D	57.5° H 45° V
<b>Sensor</b>	RGB Profundidad	RGB Profundidad	RGB Profundidad
<b>Flujo de datos color</b>	(640×480)32-bit:30fps	VGA(640x480) : 30fps QVGA(320x240):60fps	VGA(640×480): 30fps
<b>Flujo de datos profundidad</b>	(320×240)16-bits 30fps	(640x480) : 30fps	(640×480): 30fps
<b>Sistema de audio</b>	Micrófono multiarray (4) – 48 Hz	N/A	Micrófono multiarray (2)
<b>Sistema de Seguimiento</b>	Tecnología PrimeSense infra-red	Tecnología PrimeSense infra-red	Tecnología PrimeSense infra-red

Al evaluar en comparación los tres dispositivos, las especificaciones y los apartados que ofrecen, son relativamente similares, no obstante para el desarrollo e implementación del aplicativo, se opta por el uso del Kinect, debido a todo el extenso contenido y la completa documentación que se puede obtener de él, siendo así, se hará uso del dicho dispositivo su primera versión y su respectivo SDK en la versión 1.8, de igual manera se utilizara el IDE Visual Studio en su

<sup>46</sup> MICROSOFT, Kinect sensor for Xbox 360 components. Internet ([support.xbox.com/en-US/xbox-360/kinect/kinect-sensor-components](http://support.xbox.com/en-US/xbox-360/kinect/kinect-sensor-components))

<sup>47</sup> ASUS, Xtion PRO LIVE. Internet: ([asus.com/latin/Multimedia/Xtion\\_PRO\\_LIVE/specifications](http://asus.com/latin/Multimedia/Xtion_PRO_LIVE/specifications))

<sup>48</sup> PRIMESENSE, PrimeSense 3D Sensors. Internet: ([i3du.gr/pdf/primesense.pdf](http://i3du.gr/pdf/primesense.pdf))

versión 12 junto al Framework .Net<sup>49</sup> en la versión 4.5, esto debido a que el dispositivo se integra de forma nativa con dicho entorno de desarrollo, también cabe resaltar que el aplicativo en su totalidad será desarrollado en el lenguaje de programación C#.

El dispositivo Kinect se situara a una distancia aproximada de 70 centímetros sobre el suelo, y con un ángulo de inclinación entre 9 y 11 grados, con ello el Kinect se localizara a el nivel de la cintura del usuario y captara específicamente el torso, además de ello la distancia con respecto al usuario se disminuirá de 1.2 – 3.5 metros correspondientes al rango de visión del Kinect a 1.5 – 2.0 metros, para crear una constante de universalidad al momento de captar las señas, es decir que la seña siga siendo la misma sin importar el usuario que la realice.

En el apartado estético, Visual Studio nos ofrece dos posibilidades a nivel de interfaz gráfica, WPF o WinForms, de los cuales se utilizara Windows Presentation Foundation, debido a que otorga más flexibilidad al crear los componentes gráficos y controles de usuario haciendo uso del lenguaje de etiquetado XML, no obstante se integrara a su vez con la librería visual MahApp.Metro<sup>50</sup> en su versión 1.1.2 la cual nos entrega variedad de controles personalizados basados en el diseño de Windows Phone y Windows 8; también se hace uso de la librería WpfAnimateGif<sup>51</sup> en su versión 1.4.12, que ofrece el elemento gráfico para reproducir imágenes en formato GIF, por último el control MediaElement<sup>52</sup> que permite realizar la reproducción de videos en formato MP4 .

Para el siguiente punto a considerar, relacionado al procesamiento o filtrado de imágenes, se evaluaron diferentes opciones como lo son OpenCv<sup>53</sup>, EmguCv<sup>54</sup> y Aforge.Net<sup>55</sup>, y debido a que el aplicativo trabaja sobre el Framework .Net y está codificado con el lenguaje C#, se opta por trabajar con el Framework Aforge.Net en la versión 2.2.5. De igual manera sucede con el componente que otorga los modelos de inteligencia artificial y elementos estadísticos, para el aprendizaje y el reconocimiento de las señas, el Framework Accord.Net<sup>56</sup> el cual se utilizara en la versión 2.15.

### **3.2 Cadena de Markov para el análisis de patrones**

Accord.Net, provee las herramientas necesarias para realizar los respectivos procesos de aprendizaje automático de las diferentes señas con las que el usuario va a interactuar, en integración con Aforge.Net, que utilizando visión artificial, y

---

<sup>49</sup> MICROSOFT, .Net. Internet ([www.microsoft.com/net](http://www.microsoft.com/net)).

<sup>50</sup> Mahapps.Metro. Internet: ([www.mahapps.com/](http://www.mahapps.com/))

<sup>51</sup> GITHUB, WpfAnimatedGif. Internet: ([github.com/thomaslevesque/WpfAnimatedGif](https://github.com/thomaslevesque/WpfAnimatedGif))

<sup>52</sup> MICROSOFT, MediaElement Class. Internet: ([msdn.microsoft.com/en-us/library/system.windows.controls](https://msdn.microsoft.com/en-us/library/system.windows.controls))

<sup>53</sup> OpenCv. Internet: ([www.opencv.org](http://www.opencv.org))

<sup>54</sup> Emgu. Internet: ([www.emgu.com](http://www.emgu.com))

<sup>55</sup> Aforge.Net. Internet: ([www.aforgenet.com](http://www.aforgenet.com))

<sup>56</sup> Accord.Net. Internet: ([accord-framework.net](http://accord-framework.net))

combinada con el procesamiento de señales, inteligencia artificial y diferentes métodos estadísticos, realiza el reconocimiento de patrones. Dentro de este grupo de herramientas Accord.Net nos facilita máquinas de aprendizaje, redes neuronales, algoritmos genéticos, árboles de decisiones, cadenas de Markov y modelos Gaussianos, todos completamente válidos para efectuar el reconocimiento de patrones dentro del aplicativo, aunque para la implementación del mismo se hará uso de las librerías correspondientes a las cadenas de Markov.

Este modelo es una herramienta para analizar el comportamiento y el desarrollo de determinados procesos estocásticos, esto es, un proceso que evoluciona a lo largo del tiempo en torno a un conjunto de estados<sup>57</sup>, dichos estados corresponden a cada una de las señas que se ingresaron y pertenecen a la cadena, además los cambios o transiciones de estados no están predeterminados, de esta manera la nueva observación o estado será cualquier elemento del alfabeto de la cadena, que es la que especifica todos los elementos asociados a ella, influyendo en las probabilidades de transición entre estados de manera positiva sobre el sistema (decisión para reconocer el patrón).

Este alfabeto es definido dependiendo del módulo con el que el usuario este trabajando (Anexo D), y de esta manera poderle adjuntar secuencias de puntos u observaciones a los elementos del alfabeto, para crear el correspondiente patrón asociado a la seña y posteriormente entrenar la cadena, mediante el componente *MultivariateNormalDistribution*<sup>58</sup> que estima los valores correspondientes a los cambios de estados (señas) entre la cadena.

### 3.3 Sensores

Antes de efectuar dichos procesos, es necesario realizar las capturas de las señas, mediante las cámaras y los sensores que nos facilita el Kinect, en primera instancia encontramos la cámara color, la cual nos entrega una captura de video a una resolución (640x480) a 30fps, las imágenes que se obtienen mediante esta cámara, están conformadas por el modelo de colores primarios de luz (RGB), el correspondiente flujo de datos, estará habilitado durante toda la ejecución del aplicativo, debido a que es la única imagen que el usuario vera proveniente del Kinect.

Después encontramos el sensor de profundidad que gracias a sus dos cámaras de infrarrojos (IR), entrega datos de profundidad de las imágenes captadas; con dichos valores es posible hacer una recreación de una nueva imagen RGB que muestre los datos de profundidad, designando colores a los pixeles teniendo en cuenta las respectivas distancias al sensor (Figura 9).

---

<sup>57</sup> Métodos Estadísticos en Ciencias de la Vida, CADENAS DE MARKOV, p 1-8.

<sup>58</sup> Accord.Net, HiddenMarkovModel. Internet: ([accord-framework.net/docs](http://accord-framework.net/docs))

Figura 9. Imagen RGB e imagen profundidad

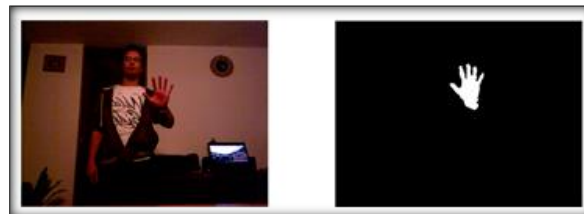


Para recrear la imagen de profundidad en aplicativo, se analizan las distancias de los pixeles de la imagen, con respecto al valor de profundidad de la mano derecha (Tabla 6), y de esta manera poder aislar la figura de la mano del resto de la imagen (Figura 10).

Tabla 6. Asignación del color para los datos de profundidad

Distancia del pixel	Valor de referencia	Color asignado
Superior a	Profundidad mano derecha - 300	0 (Negro)
Inferior a	Profundidad mano derecha + 50	0 (Negro)
Entre	Profundidad mano derecha – 300 y profundidad mano derecha + 50	255 (Blanco)

Figura 10. Imagen profundidad con colores asignados

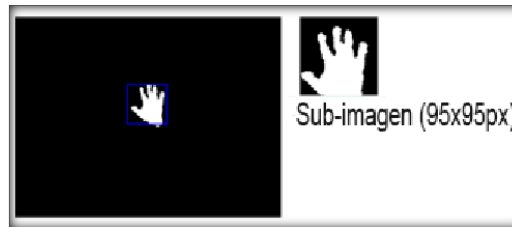


### 3.4 Procesamiento de imagen

Para efectuar el filtrado de imagen se opta por la imagen de profundidad, debido a que una imagen RGB, no otorga los niveles adecuados de contraste entre la mano y el fondo, si no existen óptimos niveles de luz en la escena, los cuales deben ser modificados dependiendo diferentes factores como la luz ambiente, fondo plano, el color del fondo, dificultando la extracción de la figura de la mano, por este motivo

se parte de la imagen de profundidad realizando un recorte de la misma creando una sub-imagen con dimensión de 95x95 píxeles (Figura 11), para realizarle el proceso de filtrado.

Figura 11. Sub-imagen profundidad mano derecha



Una vez se haya realizado la captura y el recorte de la imagen de profundidad se hace uso de la librería *Imaging.Filters*<sup>59</sup> que hace parte del Framework Aforge.Net, para efectuar el respectivo procesamiento de imagen.

El primer paso es convertir la imagen a escala de grises, para ello se hace uso del componente *Grayscale*<sup>60</sup>, que recibe 3 parámetros, que corresponden a la intensidad de los colores de luminosidad (rojo, azul, verde), para el respectivo caso se utilizan los niveles los referentes a la intensidad de luz que el ojo humano percibe<sup>61</sup>.

- Rojo 30%
- Verde 59%
- Azul 11%

Una vez la imagen ha sido puesta a escala de grises, se utiliza el componente *Threshold*<sup>62</sup>, este filtro se encarga de binarizar la imagen, tomando como referencia el valor de umbral especificado, el cual corresponde a 200, este proceso compara cada uno de los píxeles de la imagen; los pixeles que poseen intensidades iguales o superiores al valor de umbral se convierten a píxeles blancos y resto de los píxeles con intensidades por debajo del valor umbral se convierten a píxeles negros creando la nueva imagen binarizada.

Posteriormente, después del proceso de binarización, se realiza el contorneado de la imagen con ayuda del componente *CannyEdgeDetector*<sup>63</sup>, el cual analiza los cambios de color entre los píxeles, otorgando una nueva imagen con el contorno definido de la mano (Figura 12).

<sup>59</sup> Aforge.Net, *Imaging.Filters namespace*. Internet: ([www.aforgenet.com/framework/docs](http://www.aforgenet.com/framework/docs))

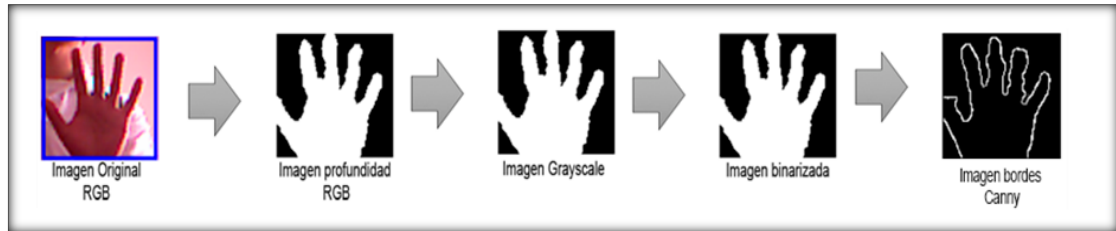
<sup>60</sup> Aforge.Net, *Grayscale Class*. Internet: ([www.aforgenet.com/framework/docs](http://www.aforgenet.com/framework/docs))

<sup>61</sup> GALARZA, osvaldo y KLEIN, roberto. "capítulo vi-señal de luminancia y crominancia". Mendoza, Argentina: Cátedras Universidad de Mendoza, facultad de ingeniería.

<sup>62</sup> Aforge.Net, *Threshold Class*. Internet: ([www.aforgenet.com/framework/docs](http://www.aforgenet.com/framework/docs))

<sup>63</sup> Aforge.Net, *CannyEdgeDetector Class*. Internet: ([www.aforgenet.com/framework/docs](http://www.aforgenet.com/framework/docs))

Figura 12. Procesamiento de imagen



### 3.5 Extracción de puntos

El primer método para obtener las secuencias de puntos corresponde a las señas dactilológicas, haciendo uso de la imagen bordes Canny, en la cual se realiza un barrido pixel por pixel de izquierda a derecha y de arriba hacia abajo extrayendo los puntos (X, Y) que correspondan a el color blanco (#FFF), creando así un listado de puntos, referentes a la imagen.

El segundo método para la extracción de puntos, correspondiente a las señas no dactilológicas, que con la ayuda de los datos de esqueleto proporcionados por Kinect, se realiza la respectiva captura de las articulaciones de interés, que corresponden a los brazos y la cabeza (Figura 13, Tabla 7), esto con el fin de reconocer únicamente el torso del usuario, y se extraer los puntos (X, Y) de las articulaciones de la mano derecha e izquierda. (Figura 14), creando así el listado de puntos.

Figura 14. Articulaciones procesadas

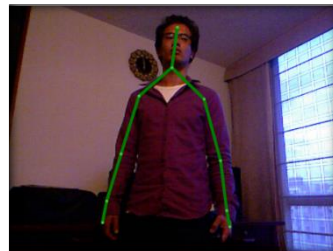


Figura 14. Articulaciones de las manos



Tabla 7. Articulaciones capturadas

Nombre Joint	Nombre Joint
<a href="#">JointType.Head</a>	<a href="#">JointType.WristLeft</a>
<a href="#">JointType.ShoulderCenter</a>	<a href="#">JointType.HandLeft</a>
<a href="#">JointType.ShoulderLeft</a>	<a href="#">JointType.ElbowRight</a>
<a href="#">JointType.ShoulderRight</a>	<a href="#">JointType.WristRight</a>
<a href="#">JointType.ElbowLeft</a>	<a href="#">JointType.HandRight</a>



Dichas extracciones de puntos y generación de colecciones, sucede siempre y cuando el usuario habilite esta función, que para ambos casos es un espacio asignado dentro de la imagen RGB. Para el caso de las señas dactilológicas es habilitado por la mano derecha, en el espacio de reconocimiento, que está situado a la altura del hombro derecho, este espacio posee unas dimensiones de 95x95 pixeles, para hacer equivalencia a la sub-imagen de profundidad.

Y para el caso de las señas no dactilológicas, el espacio para habilitar y detener la extracción de puntos, se sitúa a la altura de la cadera del usuario al costado izquierdo, este espacio posee unas dimensiones de 50x50 pixeles, y para interactuar con él, la mano izquierda es la designada. Para los dos casos el tiempo de carga se asignó a dos segundos desde el momento que se sitúa la mano en el correspondiente espacio.

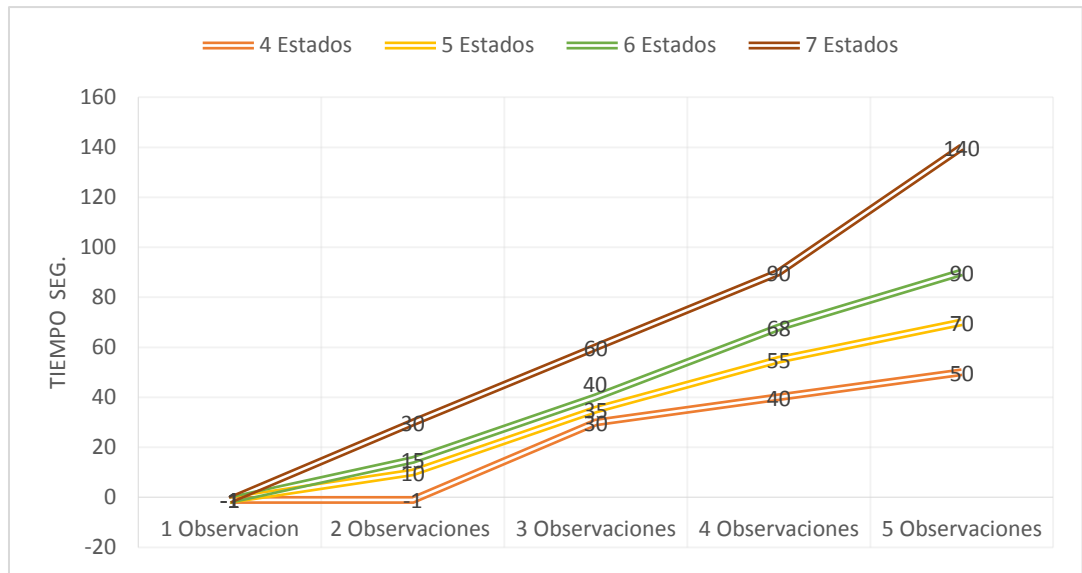
### **3.6 Creación, entrenamiento y reconocimiento de las señas**

A continuación, se utiliza la colección de puntos, y se asocia a un elemento del alfabeto de la cadena de Markov y se define si corresponde a señas dactilológicas o no, de la misma manera para cada elemento del diccionario se le debe asociar diferentes muestras, observaciones o listado de puntos para la misma seña, esto con el fin que la cadena realice la diferenciación de cada patrón; otra variable al momento de crear los patrones dentro de la cadena, es la cantidad de estados o señas que posee (Anexo D).

Para llegar al punto que se considera óptimo con respecto a tiempos de carga versus eficiencia, se realizó un proceso previo donde se evalúan diferentes señas, agrupándolas en 4, 5, 6 y 7 estados, verificando su comportamiento y tiempos de carga con 1, 2, 3, 4 y 5 observaciones dentro de la cadena de Markov, este proceso se realizó mediante la técnica de entrenamiento supervisado, la cual consiste en otorgar un conjunto de entrenamientos, realizar el aprendizaje y medir la eficiencia del resultado.

Con el conjunto de datos obtenido (Grafico 2), se llega a la conclusión que la mejor distribución corresponde a una seña con tres observaciones compuesta por 4, 5 o 6 estados, con valores entre 30 y 40 segundos de carga por cadena.

Gráfico 2. Tiempo promedio de carga con respecto a colecciones y estados de la cadena de Markov



Debido a la cantidad de estados en una cadena, se opta por hacer uso de diferentes cadenas de Markov para poder abarcar cada una de las diferentes señas correspondientes a cada módulo (Anexo D), y una vez creadas las cadenas con estos valores, se exportan sus respectivos arreglos de puntos u observaciones a un archivo XML (Anexo B), relacionando la respectiva cadena la cual pertenecen.

Una vez, ya teniendo los diferentes patrones guardados, cada vez que el usuario ingrese a un apartado o modulo en el aplicativo, automáticamente buscara el respectivo archivo XML previamente creado, y realizara el proceso de entrenamiento haciendo uso de las librerías *MultivariateNormalDistribution*<sup>64</sup> que otorgan los valores de las probabilidades de los cambios de estados en la cadena y el aprendizaje mediante la librería *BaumWelchLearning*<sup>65</sup>, la cual toma las secuencias de observaciones (Secuencia de puntos de la seña), estimando los parámetros del Modelo Markov, maximizando la probabilidad de una secuencia.

Estos valores de la cadena, son la probabilidad de transición entre estados y la convergencia de la cadena, dichos valores varían por cada proceso de aprendizaje, es decir que la cadena en cada aprendizaje aprende de diferente manera, otorgando siempre resultados distintos.

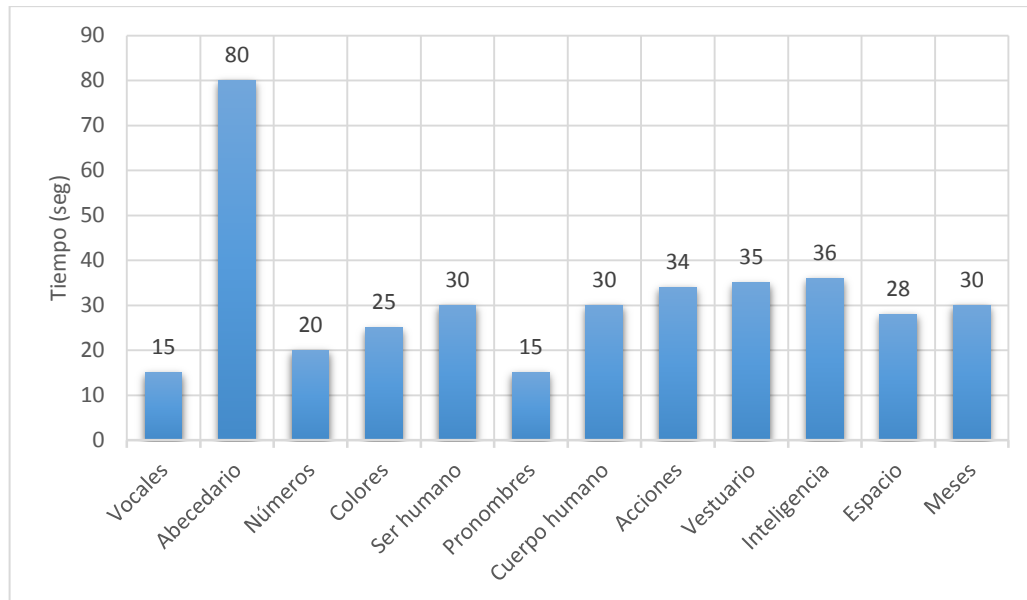
Para los entrenamientos de la cadena Markov se realizaron 10 diferentes pruebas de rendimiento para cada módulo, con respecto a los tiempos de carga de la cadena, ya teniendo todas las diferentes observaciones que la componen, donde

<sup>64</sup> Accord.Net , MultivariateNormalDistribution Internet: ([accord-framework.net/docs](http://accord-framework.net/docs)).

<sup>65</sup> Accord.Net , BaumWelchLearning Internet: ([accord-framework.net/docs](http://accord-framework.net/docs))

la cantidad de estados en la cadena afecta directamente a los tiempos de carga del entrenamiento (Grafico 3).

*Gráfico 3. Tiempos de carga para los entrenamientos por modulo*

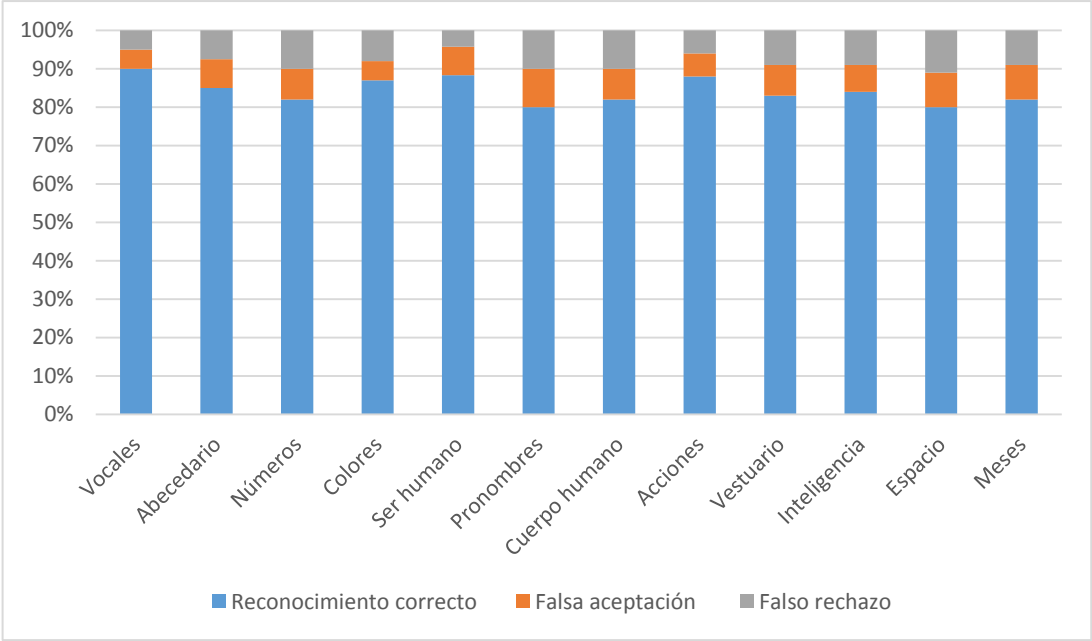


Después de realizar el proceso de entrenamiento de la cadena se realiza el reconocimiento de patrones, que corresponde a realizar una vez más el proceso de extracción de puntos, para crear una nueva secuencia, la cual es enviada a la cadena de Markov y es procesada en ella, pasando por cada uno de los estados y analizada en comparación con cada observación, dándonos como resultado el patrón con la más alta probabilidad de similitud o mayor parecido, dependiendo de los valores en el proceso de aprendizaje, puesto que los valores de convergencia y transición de los estados de la cadena varían en cada proceso de aprendizaje, haciendo que al momento de reconocer los patrones sea unas veces más precisa que otras.

Estos valores dinámicos afectan directamente al el nivel de exactitud de la cadena, por ello se realizaron diversas pruebas para las cadenas de cada módulo con 20 diferentes entrenamientos, midiendo en comparativa los valores en que el reconocimiento fue exitoso con respecto a los valores sin éxito.

Esta medición intenta obtener el valor para las tasas de error (falsa aceptación y falso rechazo) generadas en el sistema, evaluando si afectan de forma considerable el rendimiento al momento de reconocer la señal (Grafico 4).

Gráfico 4. Porcentajes de exactitud por cada modulo



## 3.7 Diagramas UML

### 3.7.1 Diagrama de clases

El diagrama de clases muestra las clases correspondientes al sistema e incluyen sus relaciones dependiendo la construcción del aplicativo, las clases que integran MetroWindows corresponden a las asociadas a la interface gráfica (Figura 15), en segunda instancia la clase ControlKinect, es la encargada de manejar y procesar todos los flujos de datos que Kinect entrega (Figura 16), y mantiene relación directa con el componente ModeloMarkov (Figura 17), la cual es la encargada de realizar el reconocimiento de las señas ingresadas, la clase ProcesarImagen es la que realiza el respectivo filtrado de la imagen finalmente las clases Excepción, las cuales controlan los resultados inesperados del aplicativo.

Figura 15. Diagrama de clases (Interface)

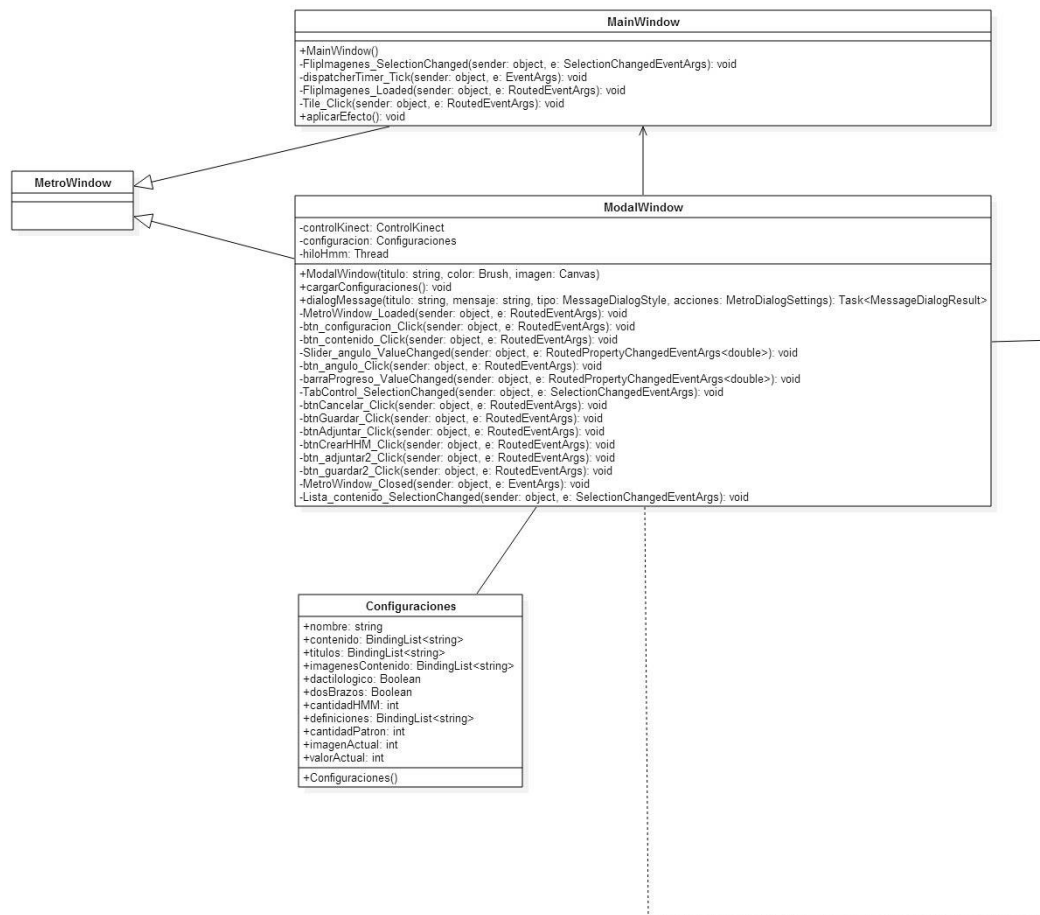


Figura 16. Diagrama de clases (Flujo de datos Kinect y filtrado de imagen)

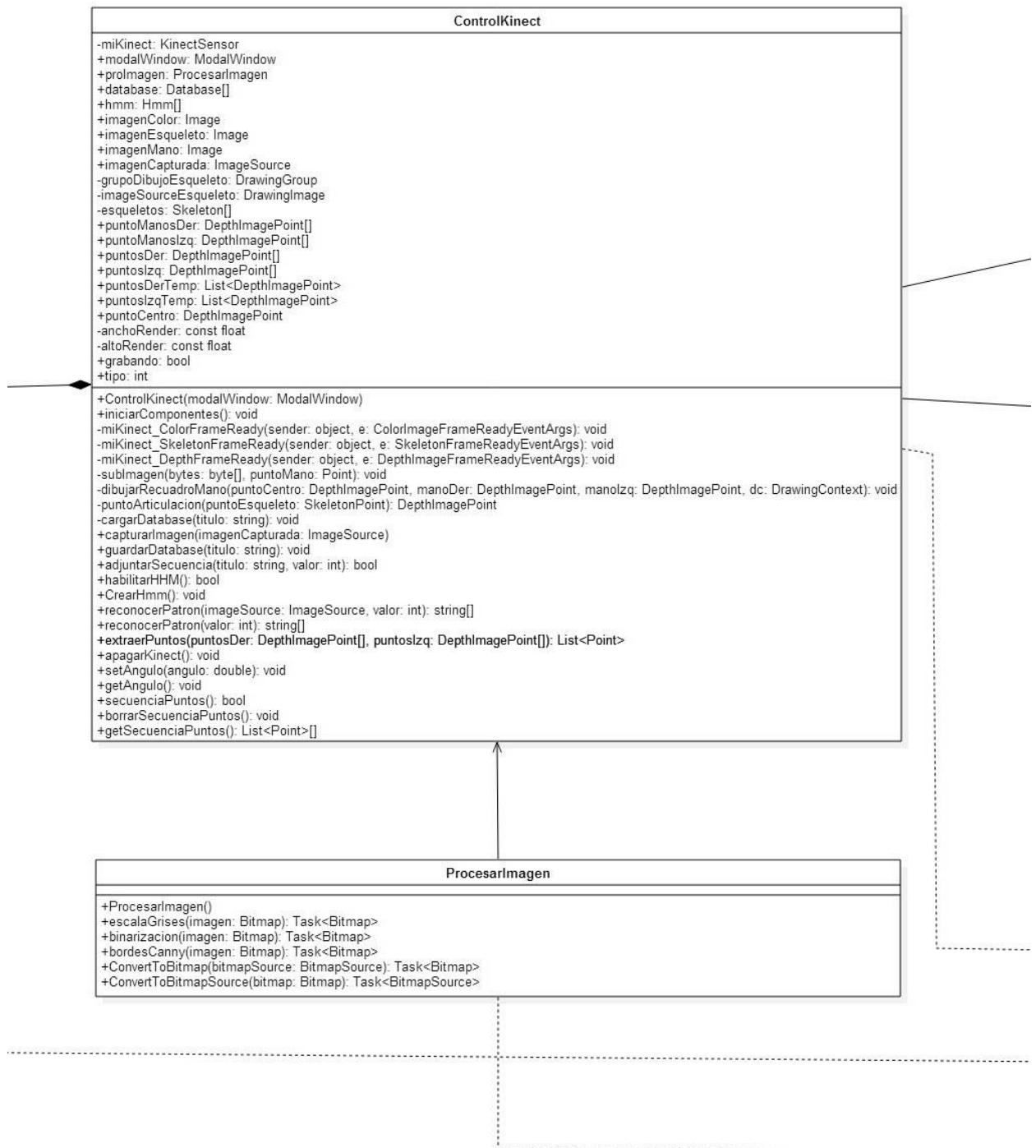
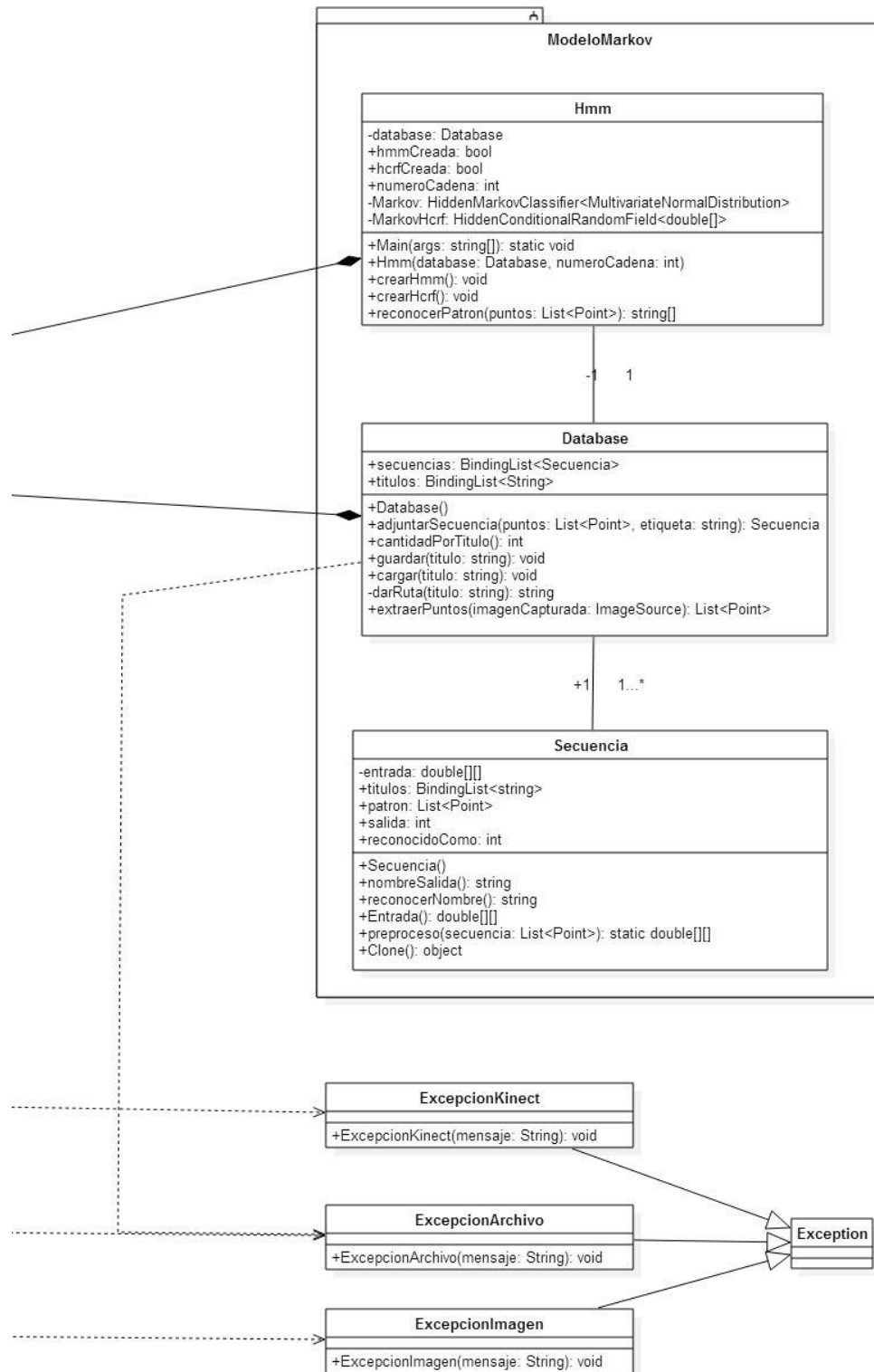


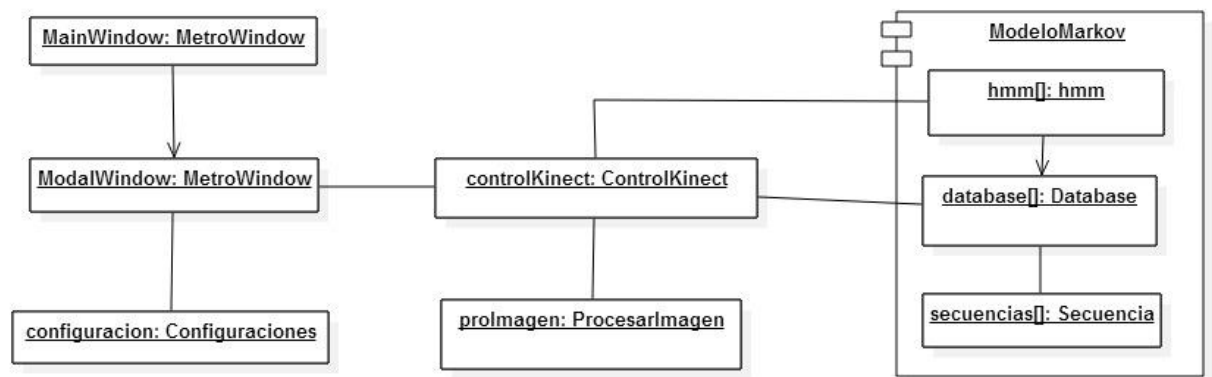
Figura 17. Diagrama de clases (Componente ModeloMarkov y Excepciones)



### 3.7.2 Diagrama de objetos

El diagrama de objetos representa las instancias de las clases dentro el aplicativo, ilustrando las relaciones entre cada objeto, de igual manera ilustra cómo está distribuido el aplicativo (Figura 18), donde se evidencia que todo lo relacionado a la cadena de Markov corresponde a un componente integrado al aplicativo.

Figura 18. Diagrama de objetos

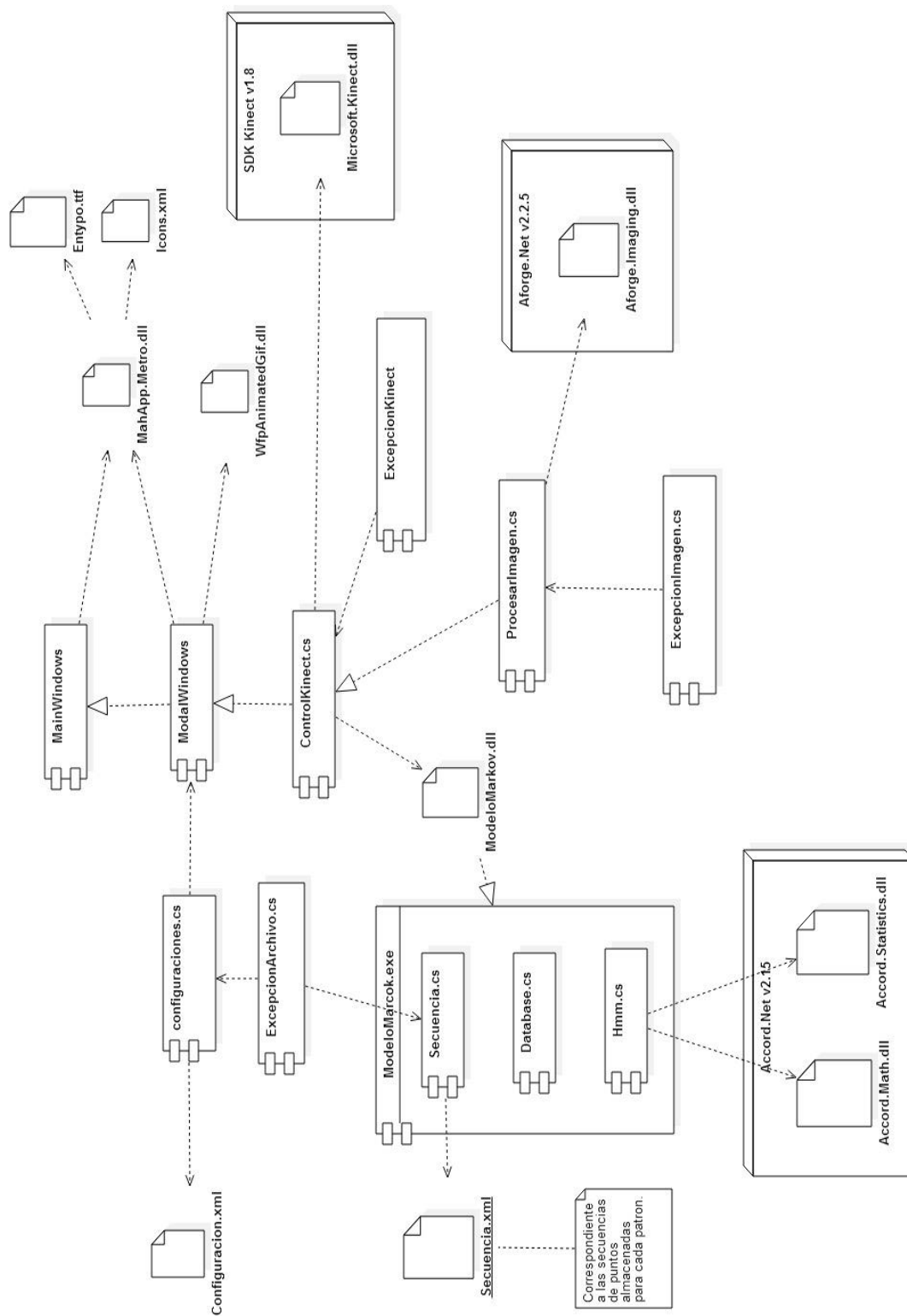


### 3.7.3 Diagrama de componentes

El diagrama de componentes muestra los elementos de diseño correspondientes al aplicativo, permitiendo visualizar con más facilidad la estructura del sistema y todos los diferentes recursos utilizados para su implementación como MetroWindows, Aforge.Net o Accord.Net, de esta manera poder describir el diseño físico del sistema (Figura 19).



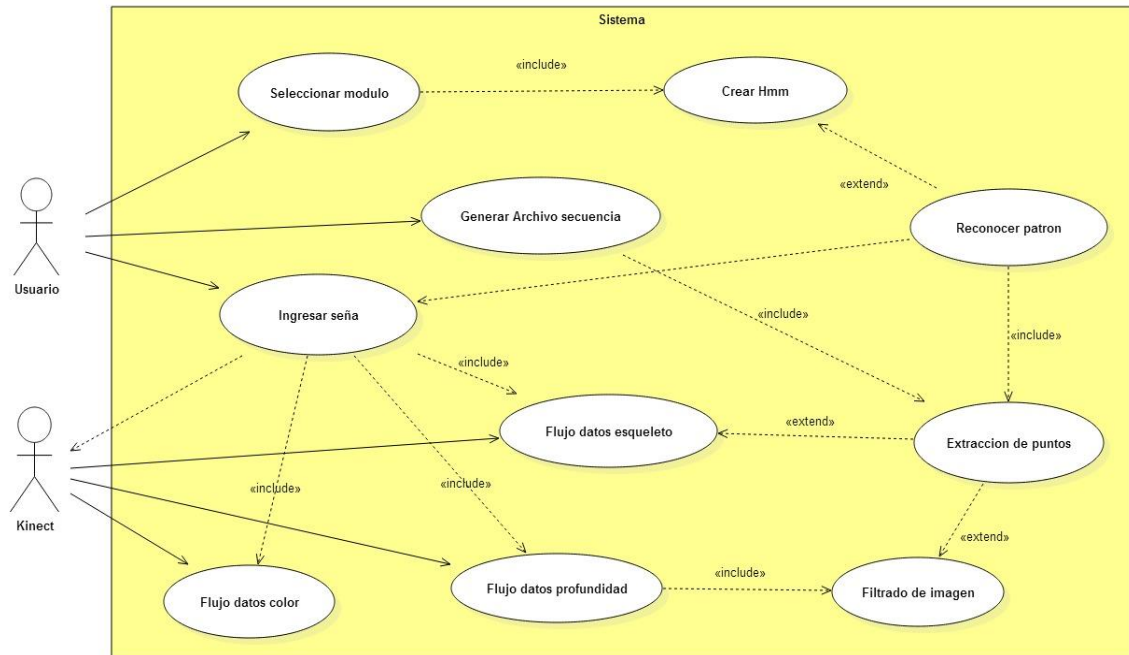
*Figura 19. Diagrama de componentes*



### 3.7.4 Diagrama de casos de Uso

El diagrama de casos de uso representa la forma en como el Usuario y el Kinect operan con el sistema, además de la ejemplificar forma, tipo y orden en cómo cada uno de los componentes del sistema interactúan entre sí, durante la ejecución del aplicativo (Figura 20).

Figura 20. Diagrama casos de uso

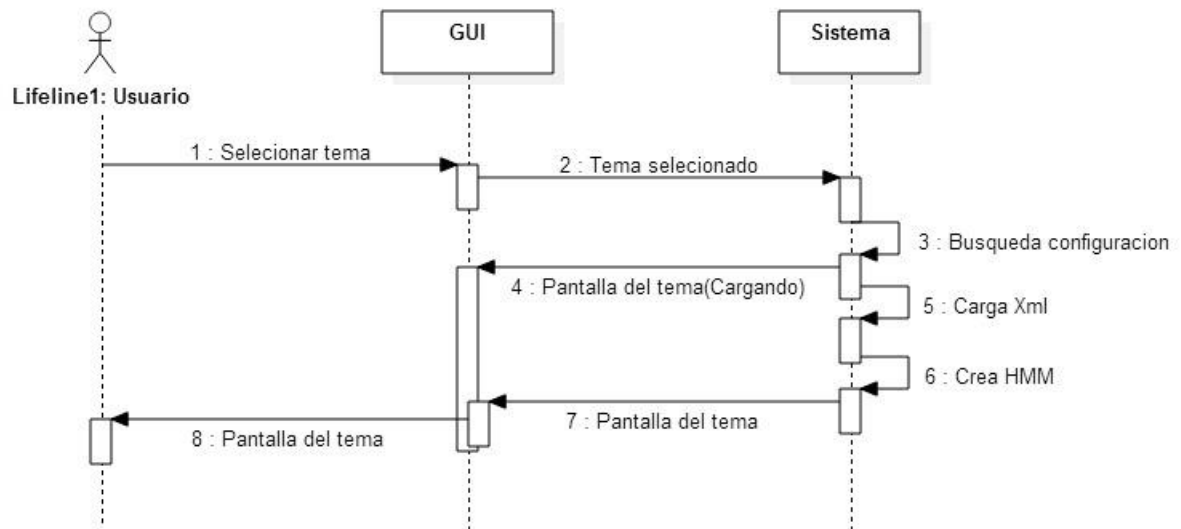


### 3.7.5 Diagrama de secuencias

El diagrama de secuencia muestra la interacción entre los componentes, ejemplificando el proceso, mostrando qué objetos se comunican con qué otros objetos y qué mensajes disparan esas comunicaciones.

El primer diagrama describe el la secuencia de eventos al momento que el Usuario, selecciona un módulo dentro el aplicativo, hasta que el sistema ejecuta la pantalla del módulo (Figura 21)

Figura 21. Diagrama de secuencia (Selección de módulo)



El segundo diagrama describe la secuencia de eventos que suceden al momento de ingresar a un módulo dactilológico, donde el usuario realiza la seña que será capturada por el Kinect, posteriormente la imagen pasara por un filtrado de imagen y finalmente el reconocimiento de la seña por medio de la cadena de Markov (Figura 22), de igual manera sucede en el tercer diagrama que describe los sucesos al ingresar a un módulo no dactilológico, en donde el usuario realiza la seña y por medio del SkeletonTracking proporcionado por Kinect se capturan los movimientos el usuario que posteriormente serán procesados en la cadena para reconocer el patrón asociado (Figura 23)

Figura 22. Diagrama de secuencia (Señas dactilológicas)

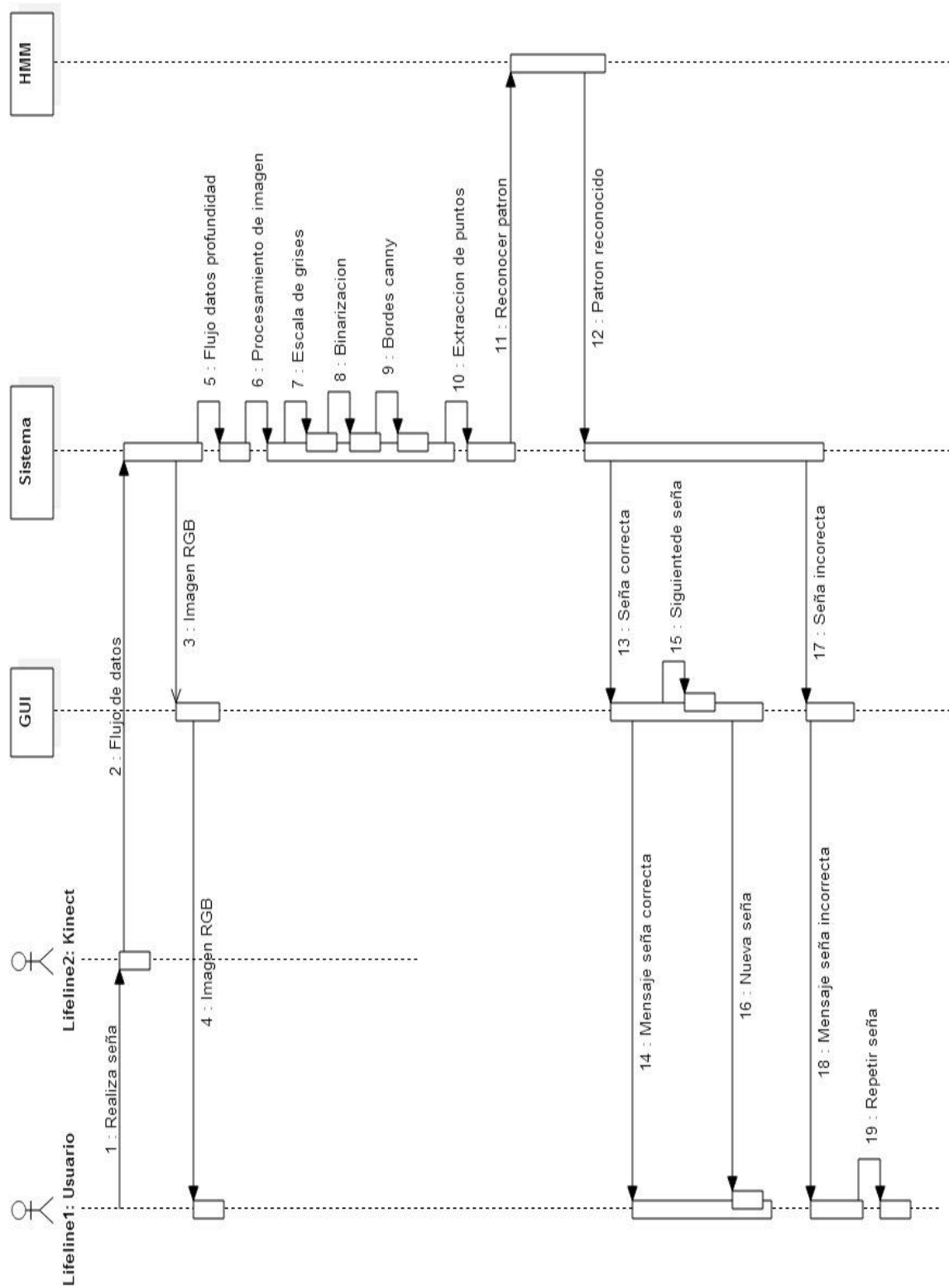
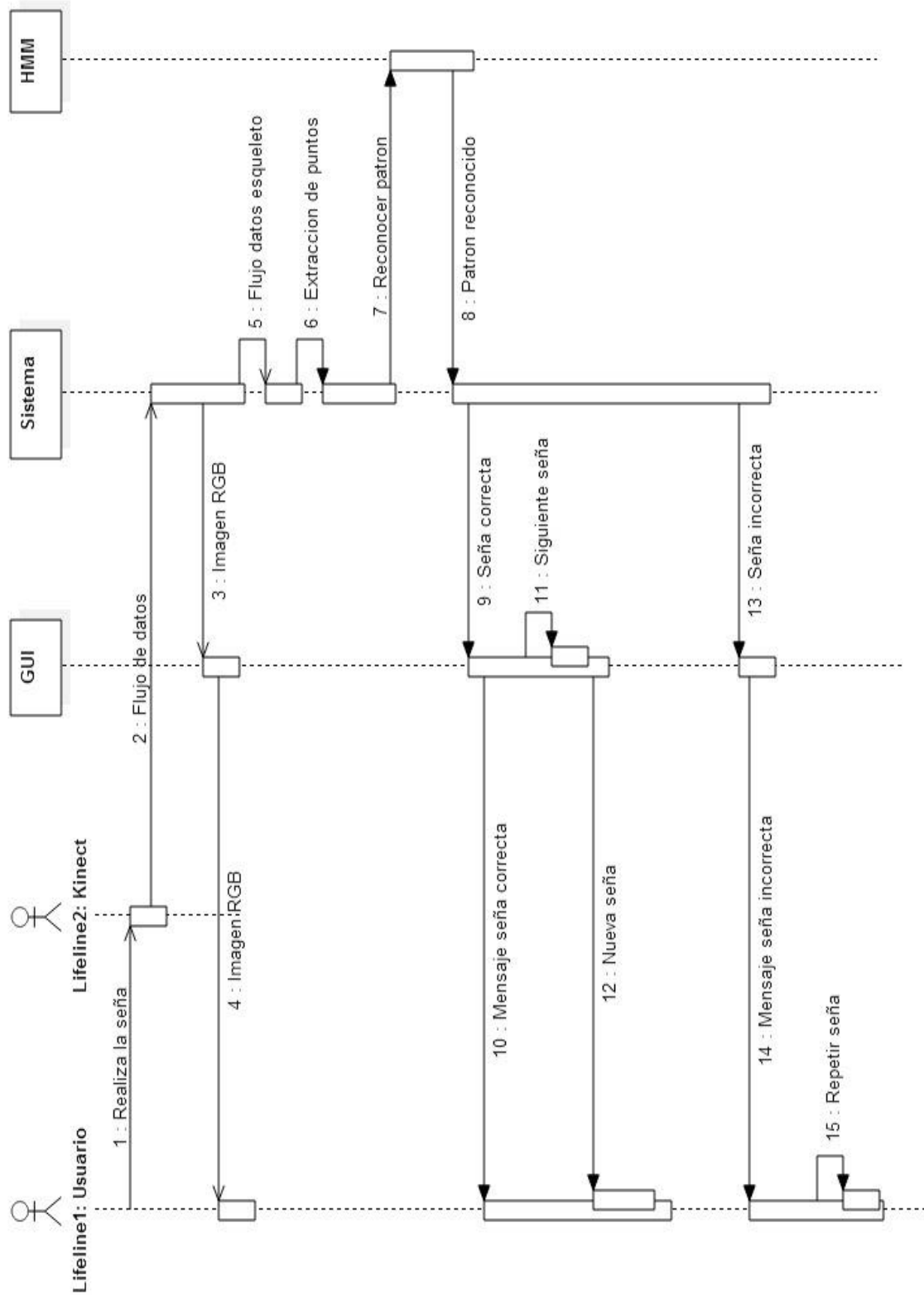


Figura 23. Diagrama de secuencia (Señas no dactilológicas)



## 4. CAPITULO 4

La validación del presente proyecto, se realizó mediante una prueba piloto en la institución académica Cemoden, con un grupo de muestra de 20 estudiantes entre los grados sexto y once, donde cada uno de ellos realizó en totalidad cada una de las señas de un módulo del aplicativo, seguido a esto se les aplicó una encuesta (Anexo E), que realizaron de manera personal teniendo en cuenta la experiencia con la aplicación.

El grupo muestra fue conformado con 10 estudiantes hombres y 10 estudiantes mujeres, elegidos de tal manera que el grupo tuviera rasgos diferenciales como lo son color de piel, estatura y densidad corporal, comprendidos entre las edades de 11 a 18 años.

La encuesta realizada fue estructurada con 12 preguntas categorizadas o de estimación, basadas en el modelo de escalamiento de Likert<sup>66</sup>, para recopilar datos respecto a las reacciones (favorable o desfavorable) de los individuos frente al aplicativo, entre las cuales el encuestado escogió la que creyó conveniente.

### 4.1 Resultados

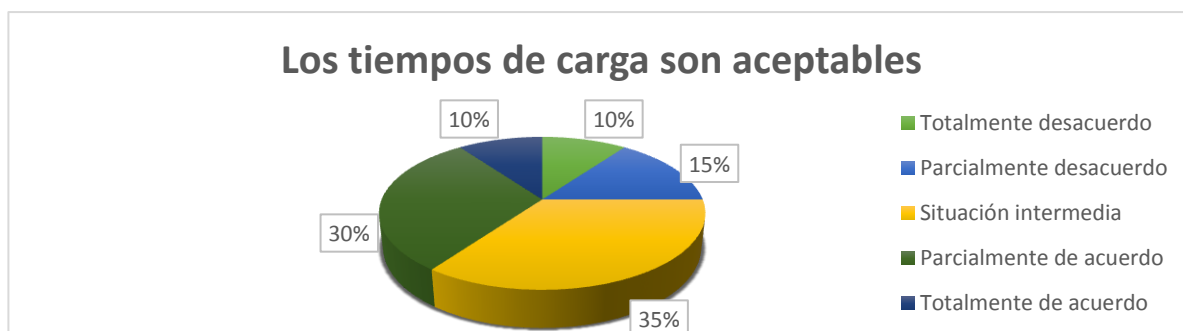
Los resultados que serán presentados a continuación hacen referencia a cada ítem de la encuesta realizada, agrupados en tres diferentes variables de análisis, la primera corresponde a los tiempos para el aplicativo. Aquí encontramos los tiempos de carga que corresponden a los tiempos relativos que toman las cadenas de Markov en aprender los diferentes patrones almacenados, creando los valores de convergencia y probabilidades para los cambios de estado, esto depende el módulo al cual el estudiante ingreso (Gráfico 5).

Debido a que módulos como Vocales y Pronombres que tienen 5 señas cada uno el promedio de tiempo en el entrenamiento es de 15 segundos, a diferencia del resto de módulos que sus tiempos varían entre 30-40 segundos con 9 señas en promedio, exceptuando a el modulo Abecedario que con 27 señas alcanza un tiempo de 100 segundos aproximadamente.

---

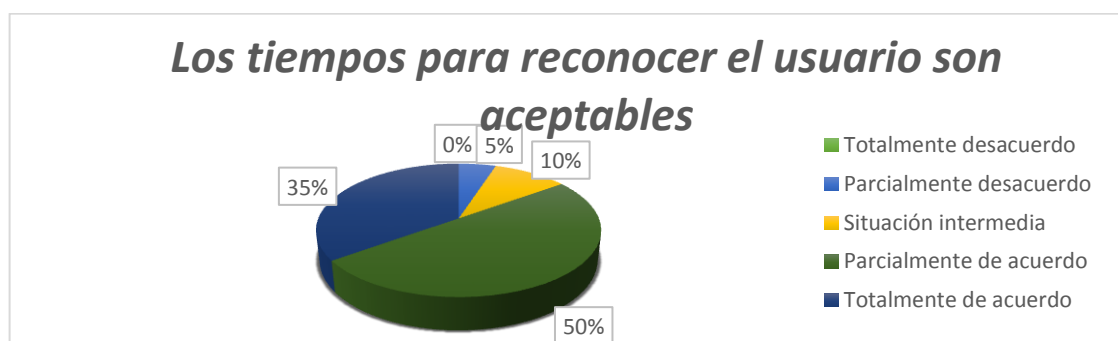
<sup>66</sup> NAVARRO, Jacinto, Modelo para enfoques de investigación acción participativa programas nacionales de formación, Maturín, Febrero año 2007. P 4-7.

Gráfico 5. Los tiempos de carga son aceptables



Seguido a esto, encontramos los tiempos en el reconocimiento de la figura del usuario, que corresponden al tiempo que el Kinect demora en reconocer el esqueleto del usuario, y posicionar ya sea el recuadro a la altura del hombro derecho para el reconocimiento de señas dactilológicas, o las figuras en torno a las manos que muestran el seguimiento de las articulaciones de las manos, los cuales variaron en un promedio de 3-5 segundos, los cuales para el 85% (17 personas) de la muestra son tiempos aceptables, mientras que un 5% (1 persona) consideran que estos tiempos no son óptimos (Gráfico 6).

Gráfico 6. Los tiempos para reconocer el usuario son aceptables



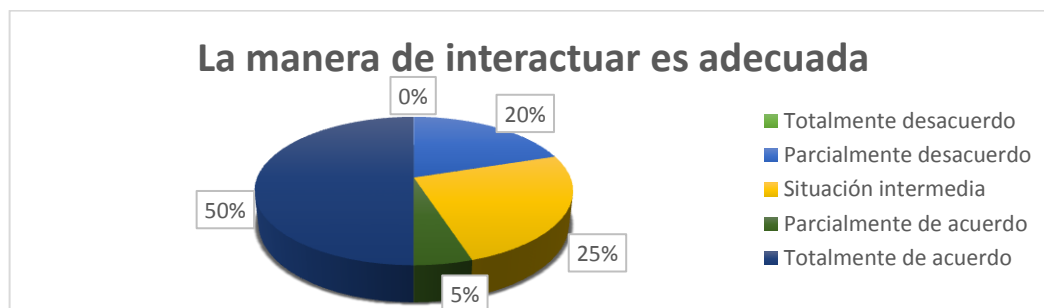
Seguido a ello, encontramos la evaluación con respecto al diseño del aplicativo y a la facilidad de uso, en primera instancia se valoró la forma en que se presentan las diferentes señas a realizar para cada módulo, siendo en formato de imagen para las señas dactilológicas, o en formato de video para señas no dactilológicas. En donde un 90% (18 personas), consideran que la manera de mostrar la forma correcta de realizar la seña es acertada, mientras un 10% (2 personas) simplemente consideran que esta bien (Gráfico 7).

Gráfico 7. La forma en que se presentan las señas es la adecuada



El siguiente ítem valúa la manera en que el usuario interactúa con el aplicativo, que puede ser de dos formas, en las señas dactilológicas se posiciona la mano derecha sobre el recuadro para la captura, o para las señas no dactilológicas donde el usuario debe activar o desactivar la captura de la seña por medio del área asignada para ello. En donde el 55% (11 personas) consideran que la manera de interacción es conveniente, a diferencia del 20% (4 personas) que piensan que la manera en cómo se implementó la interacción con el aplicativo no es la mejor (Gráfico 8).

Gráfico 8. La manera de interactuar es la adecuada

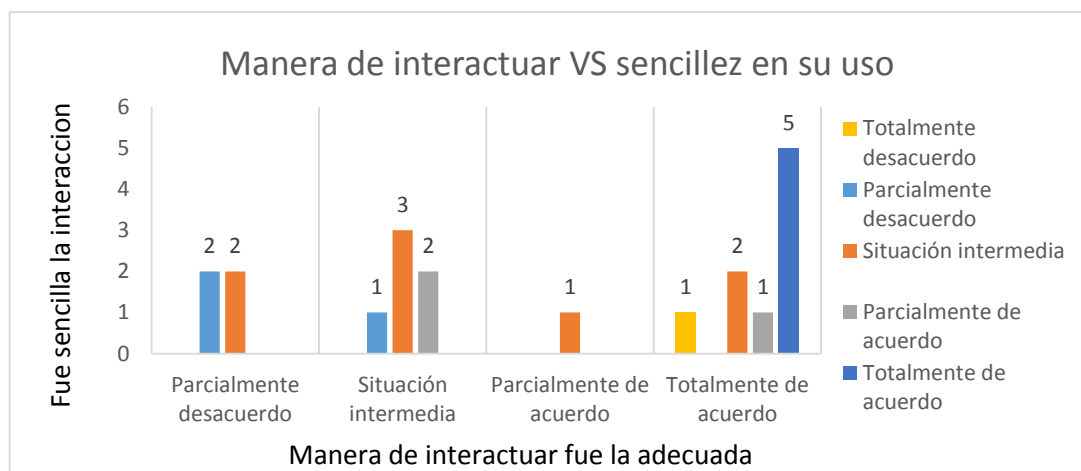


De igual manera con el anterior resultado, se segmenta dependiendo la interacción que cada individuo tuvo frente al aplicativo, validando si fue sencilla la interacción con el aplicativo o no (Gráfica 9).

En donde se ve la coherencia entre la forma de interactuar y la experiencia individual que cada sujeto tuvo al realizar las diferentes señas, dando como resultado un balance bueno acerca de la forma en que se hace la interacción con el aplicativo, aquí podemos observar que las personas que consideran que la interacción es la adecuada, el 88% (8 personas), no presentaron dificultades al momento de usar el aplicativo con excepción de un individuo.



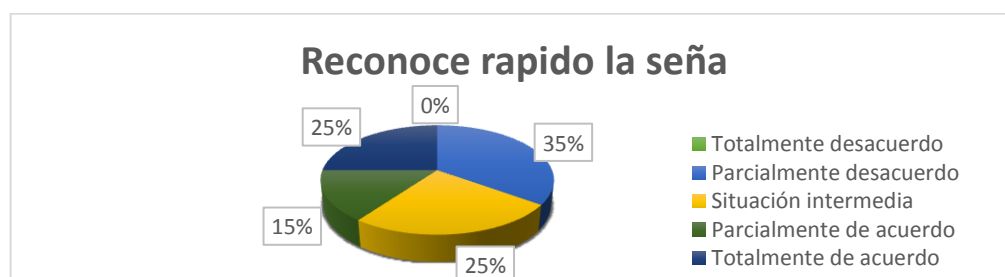
Gráfico 9. Manera de interactuar Vs sencillez en su uso



Otro elemento importante que califica al aplicativo, es la veracidad y rapidez con la que reconoce de forma efectiva la seña realizada, los dos siguientes ítems, hacen referencia a la percepción que la muestra seleccionada tuvo al respecto de la rapidez y precisión al momento de reconocer la seña.

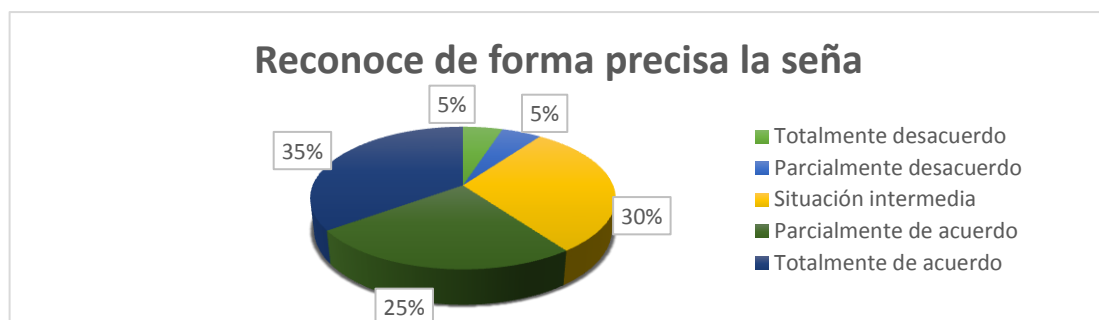
En relación a la rapidez en que se reconoce la seña, el aplicativo tarda en promedio 2 segundos en capturar, procesar y reconocer la seña realizada, dicho tiempo para el 35%(7 personas), es considerado valor poco eficiente, sin embargo el 40% (8 personas) y el 25%(5 personas) están desacuerdos que el tiempo es aceptable. (Gráfico 10).

Gráfico 10. Reconoce de forma rápida la seña



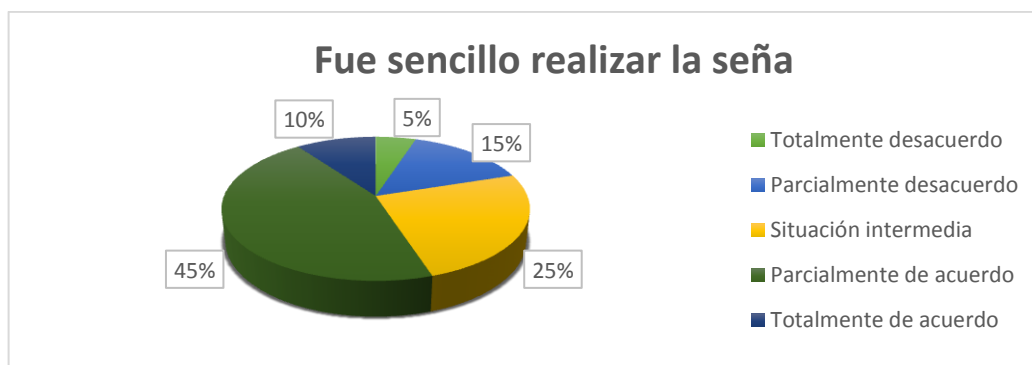
Cabe aclarar que no es suficiente que sea rápido al momento de reconocer la seña, sino que sea de una manera precisa, durante la prueba de campo los resultados obtenidos fueron bastantes buenos (Gráfico 11), puesto que el 60%(12 personas), en su interacción consideran que el reconocimiento es preciso, no obstante el 10%(2 personas) piensan que el reconocimiento debe ser más preciso.

Gráfico 11.Reconoce de forma precisa la seña



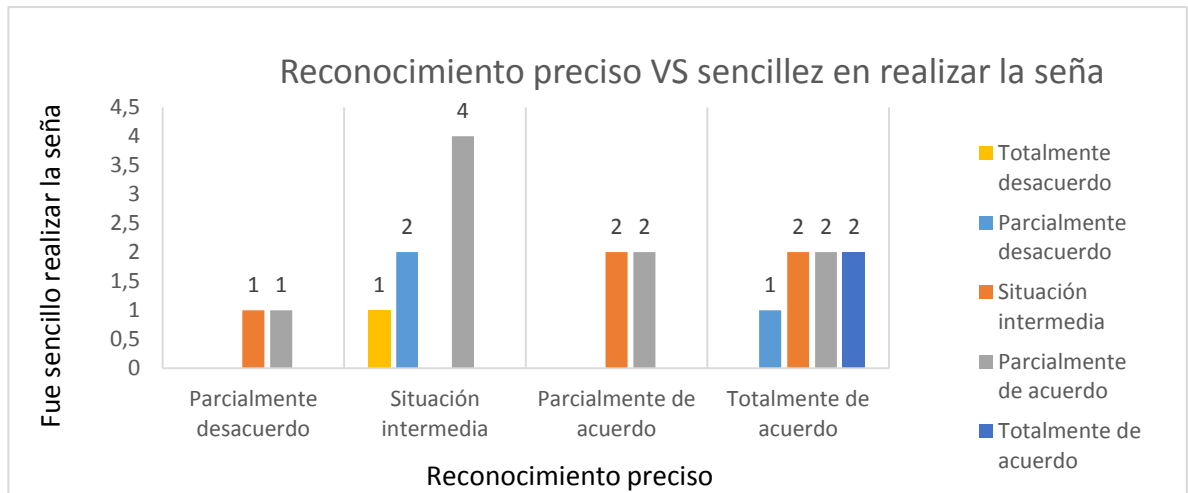
Para darnos una aproximación a la razón de los resultados al nivel de precisión al momento de realizar el reconocimiento, evaluamos la sencillez para realizar la seña (Gráfica 12), donde para el 80% (16 personas) se les facilito realizar las señas,

Gráfico 12. Fue sencillo realizar la seña



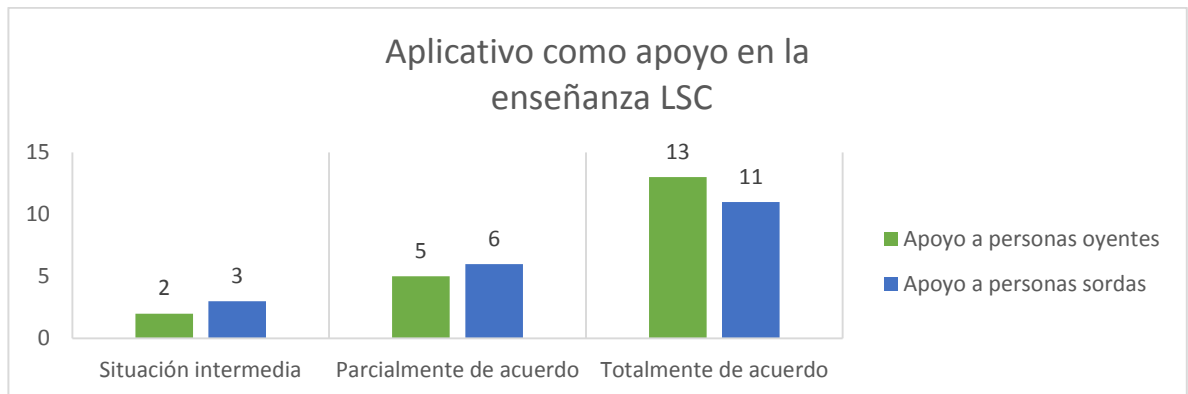
De la misma manera se hace el análisis con respecto a la precisión y la facilidad de realizar la seña, donde hay un grupo estadísticamente significativo del 20%(4 personas), que consideran que el reconocimiento no fue lo suficientemente preciso teniendo en cuenta que la realización de la seña se les hizo sencilla (Gráfica 13)

Gráfico 13. Reconocimiento preciso Vs sencillez en realizar la seña



Finalmente encontramos la concepción general del aplicativo, como un apoyo a la enseñanza de la lengua de señas, donde se evaluó si el aplicativo realmente servía como un elemento que ayuda al aprendizaje de la lengua de seña (Gráfico 14).

Gráfico 14. Aplicativo como apoyo en la enseñanza LSC



En donde se tuvieron los datos esperados, debido a que 65%(13 personas), las cuales aprobaron el aplicativo de manera significativa, al igual que el resto de individuos, que lo aprueban pero con un rango de aceptación más bajo.

En definitiva el aplicativo brinda un aporte significativo y un valor agregado a las personas oyentes que interactúan con él, mejorando sus habilidades del LSC, a medida que practican, ensayan y evalúan la realización de las señas.

## 4.2 Conclusiones

Del presente proyecto se desglosan una serie de conclusiones relevantes, no tan solo para el aplicativo como una herramienta tecnológica, que con diversos procesos de ingeniería logran desembocar en lo que aquí se llamó reconocimiento de patrones, sino también como una herramienta social que gracias a dicho proceso creativo a nivel de ingeniería, colabora en una pequeña dosis a la disminución de la llamada exclusión social hacia la comunidad sorda colombiana.

El aplicativo realiza diferentes procesos para cumplir su objetivo de reconocer señas LSC, entre los cuales se hace hincapié en el procesamiento de imágenes o filtrado de imagen, debido a que en el presente proyecto se sacrificó un tanto la eficiencia en las señas dactilológicas, para eliminar las variables de luminosidad y contraste que jugaban un papel fundamental en el filtrado tipo *Threshold* y el *CannyEdgeDetector*, movimiento acertado, porque de no haber sido así, cada vez que el aplicativo era puesto en marcha, era necesario realizar una calibración con respecto al ángulo del Kinect y modificar cada vez la escena para que los niveles de luz fueran óptimos.

Hay que tener en cuenta que los valores de convergencia de cada uno de los estados de la cadena de Markov, están ligados estrechamente con las secuencias de entrenamiento, es por ello que dos señas diferentes con muy parecidos agrupamientos de puntos, en sus tiempos de entrenamiento crezcan considerablemente, a tal punto que la cadena no sea capaz de converger a dos patrones diferentes.

Por ello es necesario jugar con la cantidad de observaciones de entrenamiento, buscando el punto óptimo entre eficiencia y tiempos de carga, puesto que estas variables son directamente proporcionales a la cantidad de entrenamientos. Por este motivo se hizo uso de tres entrenamientos, y el valor de aceptación del aplicativo se consideró en un 70%, superado levemente en la prueba piloto por un 78% de eficiencia.

Aunque los métodos y la manera con la que el usuario interactúa, son en cierta medida cómodos y sencillos de usar, cabe aclarar que los usuarios que realizan interacciones con la aplicación en unos primeros acercamientos, se les hace confusa la forma de usar y por dicha razón es ineficiente el reconocimiento. Solo hasta el momento que el usuario tiene claro el funcionamiento de las señas dactilológicas o la manera de activar la captura de datos para las no dactilológicas, el proceso se facilita para el usuario de manera considerable.

El ejemplificar las diferentes señas mediante la asociación de imagen/video con el término, permite al usuario aprender de una manera didáctica, debido a que se

hace uso de los procesos cognitivos relacionados al aprendizaje con apoyo visual, mejorando la destreza del usuario al momento de realizar la seña.

Teniendo en cuenta los resultados de las encuestas y la experiencia en la prueba piloto, es objetivo decir que el aplicativo cumple de manera aceptable con las expectativas creadas en torno al mismo, haciendo que este sea un elemento considerable al momento de dar soporte y apoyo al aprendizaje de la lengua de señas por parte de los oyentes, haciendo que jueguen un papel en la comunidad sorda colombiana.

#### **4.3 Futuras líneas de investigación**

Con el objetivo de mejorar y optimizar el aplicativo realizado, se propone realizar un sistema de login en donde cada usuario del aplicativo pueda ingresar a su cuenta, viendo estadísticas, resultados y avances con respecto a su proceso de aprendizaje de la lengua de señas.

También se deja a consideración la implementación de la presente metodología haciendo uso del dispositivo Kinect en su versión 2.0, utilizando las diferentes mejoras que el nuevo SDK proporciona, y de esta manera poder mejorar la eficiencia con respecto a las cámaras, debido a que el nuevo dispositivo lleva consigo un sensor infrarrojo y el seguimiento de articulaciones de la mano.

De igual manera se abre la posibilidad a realizar diferentes modelos de reconocimiento de patrones, para la integración con el aplicativo, y de esta manera evaluar con elementos contundentes, que diferentes algoritmos, o modelos matemático y estadísticos relacionados con inteligencia artificial, como lo son las máquinas de aprendizaje, las redes neuronales, los algoritmos genéticos, los árboles de decisiones o los modelos Gaussianos, entre otros, pueden otorgar en mayor medida una mejor eficiencia y reducidos tiempos de carga, también reducciones considerables en los falsos reconocimientos.

Otro factor importante para el mejoramiento del sistema propuesto, corresponde a la implementación del reconocimiento facial (FaceTracking), para la detención de rostros, esto con el fin de ampliar el contenido de las señas que son reconocidas, debido a que se integrarían todos los gestos faciales en el aplicativo, este tipo de señas juegan un papel importante dentro del LSC. Así mismo también se deja sobre el papel, la implementación del componente HandTacking, que complementa la función de seguimiento de la mano que actualmente está presente.

También es interesante poder ver, que el aplicativo evolucione a un sistema que reconozca e interprete la estructura gramatical de la LSC, teniendo como marco

de referencia cada una de las diferentes señas de este lenguaje, para que posteriormente de un paso hacia un sistema completo de traducción de la lengua de señas Colombia a la lengua castellana.

## ANEXOS

### ANEXO A. Modelo archivo configuración. Formato XML

```
<?xml version="1.0" encoding="utf-8"?>
<ArrayOfConfiguraciones xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <!--Configuracion primera -->
    <Configuraciones>
      <!--Titulo del tema-->
        <nombre>Vocales</nombre>
      <!--Contenido del tema -->
        <contenido>
          <!--Primer contenido -->
            <string>Contenido 1</string>
            .
            .
            <string>Contenido n </string>
          </contenido>
        <!--Valores de los contenidos -->
          <titulos>
            <!--Valor contenido -->
              <string>Valor 1</string>
              .
              .
              <string>Valor n</string>
            </titulos>
          <!--Imagenes -->
            <imagenesContenido>
              <!--Imagen primer contenido-->
                <string>Ruta 1</string>
                .
                .
                <string>Ruta n</string>
            </imagenesContenido>
          <!--Valores adicionales-->
            <dactilologico>true</dactilologico>
            <dosBrazos>false</dosBrazos>
            <cantidadHMM>1</cantidadHMM>
            <cantidadPatron>5</cantidadPatron>
          <!--Ejemplo y/o definiciones de cada contenido-->
            <definiciones>
              <string>Definicion 1</string>
              .
              .
              <string>Definicion n</string>
            </definiciones>
          </Configuraciones>
        <!--Configuracion segunda -->
          </Configuraciones>
        .
        .
      </Configuraciones>
    </ArrayOfConfiguraciones>
```

## ANEXO B. Modelo para los patrones. Formato XML

```
<?xml version="1.0" encoding="utf-8"?>

<ArrayOfSecuencia xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
xmlns:xsd="http://www.w3.org/2001/XMLSchema">

<!--Secuencia 1 del patrón 1 -->
  <Secuencia>
    <!--Nombres de todos los patrones correspondientes a el HMM -->
    <titulos>
      <!--Nombre para el primer patrón-->
      <string>Nombre de la Señal 1</string>
      .
      .
      <string>Nombre de la Señal n</string>
    </titulos>
    <patron>
      <!--Puntos de la secuencia 1 del patrón 1 -->
      <Point>
        <X>x1</X>
        <Y>y1</Y>
      </Point>
      <Point>
        <X>x2</X>
        <Y>y2</Y>
      </Point>
      .
      .
      <Point>
        <X>xn</X>
        <Y>yn</Y>
      </Point>
    </patron>
    <!--Valor de la secuencia 1 del patrón 1 -->
    <salida>0</salida>
    <!--Valor reconocido por el HMM de la secuencia 1 del patrón 1 -->
    <reconocidoComo>-1</reconocidoComo>
  </Secuencia>
  .
  .
  <!--Secuencia n del patrón n -->
  <Secuencia>
    .
    .
  </Secuencia>
</ArrayOfSecuencia>
```



## ANEXO C. Especificación casos de uso

Caso de uso	Seleccionar modulo		CU1
Actores	Usuario		
Precondición	Pantalla principal abierta		
Postcondición	Pantalla del módulo seleccionado abierta		
Propósito			
Abrir la pantalla del tema correspondiente a la seleccionada por el usuario			
Resumen			
El usuario en la pantalla principal selecciona uno de los temas disponibles, y se abrirá una nueva ventana del tema seleccionado			
Curso normal			
1	Ejecuta el aplicativo	2	Se abrirá la pantalla principal
3	Selecciona temática	4	Carga el archivo configuraciones en busca del tema seleccionado (E1,E2)
		5	Ejecuta la pantalla del tema seleccionado
		6	Verifica que exista un dispositivo Kinect conectado (E3)
		7	Crea la respectiva cadena de Markov (CU2)
		8	Carga la pantalla
Excepciones			
E1	El archivo no existe (A1)		
E2	El archivo está dañado (A2)		
E3	No existe dispositivo Kinect conectado (A3)		
Cursos alternos			
A1	Se notifica que el archivo no existe, se cierra la pantalla del modulo		
A2	Se notifica que el archivo está dañado, se cierra la pantalla del modulo		
A3	Se notifica al usuario que no existe dispositivo Kinect conectado, se cierra la pantalla del modulo		

Caso de uso	Crear HMM	CU2	
Actores			
Precondición	Seleccionado un tema		
Postcondición	Cadena de Markov creada y entrenada		
Propósito			
Crear y entrenar la cadena de Markov con las respectivas secuencias asociadas			
Resumen			
Se crea la cadena de Markov, se asocian las secuencias para entrenar la cadena y posteriormente reconocer patrones en las señas			
Curso normal			
		1	Se crea la cadena de Markov
		2	Se lee el archivo XML correspondiente a dicha cadena de Markov (E1,E2)
		3	Se crea un diccionario de datos, con las secuencias en el archivo XML
		4	Se entrena la cadena de Markov
Excepciones			
E1	El archivo no existe (A1)		
E2	El archivo está dañado (A2)		
Cursos alternos			
A1	Se notifica que el archivo no existe, se cierra la pantalla del modulo		
A2	Se notifica que el archivo está dañado, se cierra la pantalla del modulo		

Caso de uso	Ingresar seña	CU3
Actores	Usuario, Kinect	
Precondición	Dispositivo Kinect conectado, pantalla del módulo abierta (CU1)	
Postcondición		
Propósito		
Ingresar una seña de forma temporal en el sistema		
Resumen		
El usuario realiza la respectiva seña y el Kinect captura los datos correspondientes a la seña		
Curso normal		
1	Realiza la seña	2 El Kinect captura los datos de la seña
		3 Se abren el flujo de datos color (CU4)
		4 Se abren el flujo de datos profundidad (CU5)
		5 Se abren el flujo de datos esqueleto (CU7)
Excepciones		
Cursos alternos		

Caso de uso	Flujo de datos color	CU4	
Actores	Kinect		
Precondición	Dispositivo Kinect conectado, pantalla del módulo abierta (CU1)		
Postcondición	Imagen RBG en la pantalla del modulo		
Propósito			
Mostrar al usuario el video que Kinect está capturando			
Resumen			
Se procesan los datos de color, para entregar una imagen RBG al usuario; esta imagen tiene una velocidad de 30fps			
Curso normal			
		1	Se abre el flujo de datos color
		2	Se adjuntan a una imagen 640x480px
		3	La imagen se muestra en pantalla
Excepciones			
Cursos alternos			

Caso de uso	Flujo de datos profundidad	CU5
Actores	Kinect	
Precondición	Dispositivo Kinect conectado, pantalla tema abierta (CU1), el modulo corresponde a señas dactilológicas	
Postcondición		
Propósito		
Crear la imagen de profundidad que Kinect está capturando		
Resumen		
Se procesan los datos de profundidad, para entregar una imagen RBG al usuario; esta imagen tiene una velocidad de 30fps		
Curso normal		
		1 Se abre el flujo de datos profundidad
		2 Se adjuntan a una imagen 640x480px
		3 se recrea la imagen profundidad con las distancias de la mano derecha
		4 Se realiza el recorte a una imagen 95x95px
		5 Se realiza el filtrado de imagen (CU6)
Excepciones		
Cursos alternos		

Caso de uso	Filtrado de imagen	CU6
Actores		
Precondición	Imagen profundidad capturada	
Postcondición	Imagen bordes Canny	
Propósito		
Crear la imagen de la mano contorneada		
Resumen		
Se toma la imagen de profundidad de realizan procesos de filtrado, escala de grises, binarización y bordes canny para contornear la imagen de la mano		
Curso normal		
		1 La imagen profundidad se filtra a escala de grises (E1)
		2 Se realiza el proceso de binarización con un filtro de 200 (E1)
		3 Se analizan los niveles de los pixeles con el método bordes Canny, con el fin de extraer el contorno de la imagen (E1)
		4 Se extraen los puntos de la imagen (CU8)
Excepciones		
E1	La imagen no es soportada (A1)	
Cursos alternos		
A1	Se aborta el proceso de filtrado, se realiza nuevamente la captura de imagen profundidad (CU5)	

Caso de uso	Flujo de datos esqueleto	CU7
Actores	Kinect	
Precondición	Dispositivo Kinect conectado, pantalla del módulo abierta (CU1), el modulo corresponde a señas no dactilológicas	
Postcondición	Entrega cada una de las articulaciones reconocidas por Kinect	
Propósito		
Obtener los datos de las articulaciones que Kinect está capturando		
Resumen		
Se procesan los datos de esqueletos, para realizar el seguimiento a las manos del usuario		
Curso normal		
		1 Se abre el flujo de datos esqueletos
		2 Se obtienen las articulaciones correspondientes al torso
		3 Se extraen los datos de las articulación de las manos
		4 Se extraen los puntos de las manos (CU8)
Excepciones		
Cursos alternos		

Caso de uso	Extracción de puntos	CU8	
Actores			
Precondición	Imagen bordes canny o flujo de datos esqueleto		
Postcondición	Secuencia de puntos de la seña ingresada		
Propósito			
Generar la secuencia de puntos			
Resumen			
Se obtiene la secuencia de puntos u observación correspondiente a la seña ingresada.			
Curso normal			
		A	Para imagen bordes canny
		1	Se recorre los pixeles de la imagen
		2	Se extraen los pixeles (X, Y), correspondientes al color blanco (#FFF)
		3	Se procesa la secuencia (A1,A2)
		B	Para datos esqueleto
		1	Se toman los puntos de las manos (X, Y)
		2	Se procesa la secuencia (A1,A2)
Excepciones			
Cursos alternos			
A1	A partir de la seña ingresada se crea el archivo XML de entrenamiento (CU9)		
A2	Se compara la secuencia de puntos, en la cadena de Markov, para reconocer el patrón (CU9)		



Caso de uso	Generar archivo	CU9
Actores	Usuario	
Precondición	Secuencia de puntos generada	
Postcondición	Archivo XML con la secuencia guardada	
Propósito		
Crear el archivo de entrenamiento para la cadena		
Resumen		
Se exportan las diferentes secuencias en un archivo XML, que contiene las secuencias de puntos asociadas a la cadena de Markov a la cual pertenece		
Curso normal		
1	Ingresa el nombre del elemento	
		2 Adjunta la secuencia de puntos a un elemento del alfabeto de la cadena (E1)
3	Elige la opción de crear archivo	4 Genera el archivo XML, con las secuencias de la cadena correspondiente (E2,E3)
		5 Se guarda el archivo satisfactoriamente
		6 Se informa al usuario que el archivo se generó de forma exitosa
Excepciones		
E1	El elemento no existe (A1)	
E2	El archivo esta corrupto o dañado (A2)	
E3	El archivo no existe (A2)	
Cursos alternos		
A1	Se informa al usuario que el elemento no es válido, se vuelve a pedir nombre del elemento	
A2	Se informa al usuario del inconveniente	

Caso de uso	Reconocer patrón	CU10	
Actores			
Precondición	Cadena de Markov creada (CU2), puntos de la nueva seña (CU8)		
Postcondición	Resultado al reconocer la seña		
Propósito			
Reconocer el patrón de la nueva seña ingresada con respecto a las señas en la cadena de Markov			
Resumen			
La cadena de Markov procesara la nueva secuencia de puntos, otorgando el valor del alfabeto a la cual corresponde la mayor coincidencia dependiendo el proceso de aprendizaje			
Curso normal			
		1	La secuencia de puntos es enviada a la cadena de Markov
		2	La cadena entrega un elemento del alfabeto
		3	Se compara el elemento reconocido con el actual
		4	Se informa al usuario si realizo la seña correcta o incorrecta
		5	Se pasa a la siguiente seña (A1)
Excepciones			
Cursos alternos			
A1	En caso incorrecto, el usuario repite la seña		

## ANEXO D. Módulos correspondientes a las señas

Las señas seleccionadas fueron tomadas del “DICCIONARIO BÁSICO DE LA LENGUA DE SEÑAS COLOMBIANA”<sup>67</sup> elaborado por Insor, siendo la guía principal para la elaboración de los patrones a partir de dichas señas.

Vocales	Abecedario			Números
<ul style="list-style-type: none"> <li>Vocal A</li> <li>Vocal E</li> <li>Vocal I</li> <li>Vocal O</li> <li>Vocal U</li> </ul>	<ul style="list-style-type: none"> <li>Letra A</li> <li>Letra B</li> <li>Letra C</li> <li>Letra D</li> <li>Letra E</li> <li>Letra F</li> <li>Letra G</li> <li>Letra H</li> <li>Letra I</li> </ul>	<ul style="list-style-type: none"> <li>Letra J</li> <li>Letra K</li> <li>Letra L</li> <li>Letra M</li> <li>Letra N</li> <li>Letra Ñ</li> <li>Letra O</li> <li>Letra P</li> <li>Letra Q</li> </ul>	<ul style="list-style-type: none"> <li>Letra R</li> <li>Letra S</li> <li>Letra T</li> <li>Letra U</li> <li>Letra V</li> <li>Letra W</li> <li>Letra X</li> <li>Letra Y</li> <li>Letra Z</li> </ul>	<ul style="list-style-type: none"> <li>Numero 0</li> <li>Numero 1</li> <li>Numero 2</li> <li>Numero 3</li> <li>Numero 4</li> <li>Numero 5</li> <li>Numero 6</li> <li>Numero 7</li> <li>Numero 8</li> <li>Numero 9</li> <li>Numero 10</li> </ul>
Colores	Ser humano	Pronombres	Cuerpo humano	
<ul style="list-style-type: none"> <li>Color</li> <li>Amarillo</li> <li>Azul</li> <li>Rojo</li> <li>Naranja</li> <li>Morado</li> <li>Café</li> <li>Verde</li> <li>Gris</li> <li>Blanco</li> <li>Negro</li> </ul>	<ul style="list-style-type: none"> <li>Persona</li> <li>Hombre</li> <li>Mujer</li> <li>Bebe</li> <li>Niño</li> <li>Joven</li> <li>Adulto</li> <li>Mayor</li> <li>Anciano</li> <li>Homosexual</li> </ul>	<ul style="list-style-type: none"> <li>Yo</li> <li>Tu</li> <li>Él / Ella</li> <li>Nosotros</li> <li>Ellos</li> </ul>	<ul style="list-style-type: none"> <li>Cuerpo</li> <li>Cabeza</li> <li>Cara</li> <li>Cuello</li> <li>Hombros</li> <li>Brazos</li> <li>Manos</li> <li>Piernas</li> <li>Pies</li> </ul>	
Vestuario	Acciones	Inteligencia	Espacio	
<ul style="list-style-type: none"> <li>Ropa</li> <li>Camisa</li> <li>Camiseta</li> </ul>	<ul style="list-style-type: none"> <li>Caminar</li> <li>Comer</li> <li>Correr</li> </ul>	<ul style="list-style-type: none"> <li>Aprender</li> <li>Atención</li> <li>Crear</li> </ul>	<ul style="list-style-type: none"> <li>Abajo</li> <li>Adelante</li> <li>Adentro</li> </ul>	

<sup>67</sup> <http://www.ucn.edu.co/e-discapacidad/Documents/36317784-Diccionario-lengua-de-senas.pdf>

<ul style="list-style-type: none"> <li>• Blusa</li> <li>• Corbata</li> <li>• Chaqueta</li> <li>• Pantalón</li> <li>• Zapatos</li> </ul>	<ul style="list-style-type: none"> <li>• Crecer</li> <li>• Descansar</li> <li>• Dormir</li> <li>• Observar</li> <li>• Oír</li> <li>• Reír</li> <li>• Respirar</li> <li>• Soñar</li> <li>• Tocar</li> </ul>	<ul style="list-style-type: none"> <li>• Descubrir</li> <li>• Dudar</li> <li>• Entender</li> <li>• Olvidar</li> <li>• Pensar</li> </ul>	<ul style="list-style-type: none"> <li>• Afuera</li> <li>• Arriba</li> <li>• Detrás</li> </ul>
Meses			
<ul style="list-style-type: none"> <li>• Enero</li> <li>• Febrero</li> <li>• Marzo</li> <li>• Abril</li> <li>• Mayo</li> <li>• Junio</li> <li>• Julio</li> <li>• Agosto</li> <li>• Septiembre</li> <li>• Octubre</li> <li>• Noviembre</li> <li>• Diciembre</li> </ul>			

## ANEXO E. Formato de encuesta

Nombre	Edad	Genero	Fecha
		M F	25-05-2015

Amablemente lo invitamos a diligenciar la presente encuesta	SI	NO
¿Usted tiene alguna discapacidad auditiva?		
¿Usted conoce o maneja la lengua de señas Colombia?		
¿Usted ha tenido contacto con alguna persona con discapacidad auditiva?		

A continuación encontrara una serie de enunciados que corresponden la efectividad, confiabilidad y veracidad del aplicativo. Para cada enunciado exprese su grado de conformidad, según su experiencia con el mismo. Utilice la siguiente escala.

1-Totalmente desacuerdo 2-Parcialmente desacuerdo 3- Situación intermedia 4-Parcialmente de acuerdo 5- Totalmente de acuerdo

	1	2	3	4	5
Los tiempos de carga en el aplicativo son aceptables					
El tiempo en reconocer la figura del usuario es aceptable					
La forma en que se presentan las señas es la adecuada					
La manera de interactuar con el aplicativo es la adecuada					
Le fue sencillo la interacción con el aplicativo					
El aplicativo reconoce de forma rápida la seña realizada					
El aplicativo reconoce de forma precisa la seña realizada					
Le fue sencillo realizar las señas del aplicativo					
El aplicativo es una buena herramienta para la enseñanza LSC a <b>oyentes</b>					
El aplicativo es una buena herramienta para apoyar el aprendizaje LSC a <b>oyentes</b>					
El aplicativo es una buena herramienta para la enseñanza LSC a <b>personas sordas</b>					
El aplicativo es una buena herramienta para apoyar el aprendizaje LSC a <b>personas sordas</b>					

Observaciones:

---



---



---



---



---



---

## BIBLIOGRAFIA

AYALA C, Jaime A, Propuesta conceptual para la elaboración de un currículo para la lengua de señas colombiana a partir de lineamientos curriculares. En: Horizontes pedagógicos. Enero-Diciembre, 2010, vol. 12. No.1. ISSN: 0123-8264

BARRETO, Alex G y CORTÉS, Yenny M. Aspectos relevantes del discurso en Lengua de Señas Colombiana (LSC). Bogotá: Universidad Nacional de Colombia.

BLOOM, L & LAHEY, M. Language development and language disorders. New York, Wiley, 1978.

CAPILLA MARTINEZ, Daniel. Sign language translator using Microsoft Kinect XBOX 360". Master tesis ERASMUS MUNDUS Masters Vibot. Knoxville, USA. Departamento de ingeniería electrónica y ciencias de la computación. 2012.

CHACÓN, Esteban, AGUILAR, Darwin y SÁENZ Fabián. Desarrollo de una Interfaz para el Reconocimiento Automático del Lenguaje de Signos. Trabajo de grado. Valle de los Chillos, Ecuador: Universidad De Las Fuerzas Armadas – ESPE. Departamento de Eléctrica y Electrónica en Redes y Comunicación de Datos. 2013.

COLOMBIA. CONGRESO DE LA REPUBLICA. Ley 324. (11, octubre, 1996). Por la cual se crean algunas normas a favor de la Población Sorda. Diario oficial. Bogotá, D.C., 1996. no. 42899.

DANG BINH, Nguyen, SHUICHI, Enokida y EJIMA, Toshiaki: Real-Time Hand Tracking and Gesture Recognition System. En: GVIP 05 Conference (19-21, Diciembre: El Cairo, Egipto). 2005.

DDHH Monthly Communicator, Trenton, NJ.

Deafness Cognition and Language Research Centre, Is sign language the same the world over?, Internet: ( <http://www.ucl.ac.uk/dcal/faqs/questions/bsl/question6>)

Dr. LAFUENTE, Antonio D., Tipos y causas de hipoacusia. Internet: (<http://sorderayvertigo.com/hipoacusias2>)

Dr. Matamala, Apuntes Neuroanatomia- UFRO: Oído. Universidad de la frontera. Temuco, Chile.

GALARZA, osvaldo y KLEIN, roberto. "capítulo vi-señal de luminancia y crominancia". Mendoza, Argentina: Cátedras Universidad de Mendoza, facultad de ingeniería. Internet: ([www.um.edu.ar/catedras/claroline/backends/download.php?cidreset=true&cidreq=1051](http://www.um.edu.ar/catedras/claroline/backends/download.php?cidreset=true&cidreq=1051))

GONZÁLEZ DEL YERRO, Asunción, Dificultades en la adquisición del lenguaje oral. Madrid: Universidad Autónoma de Madrid. 2012.

INSOR. Diccionario básico de la lengua de señas colombiana. Bogotá: 2006.

Instituto Nacional de la Sordera y otros Trastornos de la Comunicación, La pérdida de audición ocasionada por el ruido, NIH Pub. No. 09-4233, Actualizado en junio del 2009

La Morfología lingüística en Gramática. Internet: (<http://www.deperu.com/abc/gramatica/2196/la-morfologia-lingistica-en-gramatica>)

LÓPEZ, Enrique Antón. Reconocimiento automático de lenguaje de signos: Lenguaje ASL. Trabajo grado Ingeniería técnica en informática de sistemas. Barcelona: Universidad de Barcelona. Facultad de Matemáticas, 2009.

LOSSON, Oliver y VANNOBEL, Jean-Marc: Sign Specification and Synthesis. En: Gesture-Based Communication in Human-Computer Interaction: International Gesture Workshop, GW'99. Marzo de 1999, Francia, p 139-250.

MARULANDA, Andrés y SERNA, Alejandra, Conoce y Aprende el Lenguaje de Señas Colombiano. Makaia, Internet: (<http://www.makaia.org/recursos.shtml?apc=h1d1---&x=1682>)

MICROSOFT, Kinect for Windows Sensor Components and Specifications. Internet: (<http://msdn.microsoft.com/en-us/library/jj131033.aspx>.)

MICROSOFT, Skeletal Tracking. Internet: <http://msdn.microsoft.com/en-us/library/hh973074.aspx>

MICROSOFT. Kinect for Windows SDK. Internet: ([msdn.microsoft.com/en-us/library/hh855347.aspx](http://msdn.microsoft.com/en-us/library/hh855347.aspx))

O'SHANAHAN, Isabel, Enseñanza del lenguaje oral y las teorías implícitas del profesorado. Trabajo de grado. La laguna: Universidad de la Laguna. Humanidades y ciencias sociales. 1996.

OVIEDO, Alejandro. Apuntes para una gramática de la lengua de señas colombiana. Cali: Universidad del Valle-INSOR. 2000.

Rangos de audición, Internet: (<http://auditivos.lux.mx/index.php/perdida-auditiva/rangos-de-audicion>)

SAPIR, Edward, El lenguaje: Introducción al estudio del habla. 1921.

SHANTZ, Michael y POIZNER, Howard: A computer program to synthesize American Sign Language. En: Behavior Research Methods & Instrumentation. 1982, vol. 14(5), p. 467-474.

Subgerencia Cultural del Banco de la República. "Lengua y lenguaje". Internet: ([http://www.banrepcultural.org/blaavirtual/ayudadetareas/comunicacion/lengua\\_y\\_lenguaje](http://www.banrepcultural.org/blaavirtual/ayudadetareas/comunicacion/lengua_y_lenguaje)), 2015.

World Federation of the Deaf (WFD), Sign Language. Internet: (<http://wfdeaf.org/our-work/focus-areas/sign-language>)